

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-203457

(43)Date of publication of application : 30.07.1999

(51)Int.Cl.

G06T 1/00

H03M 7/30

H04N 1/41

(21)Application number : 10-017935

(71)Applicant : NEC CORP

(22)Date of filing : 14.01.1998

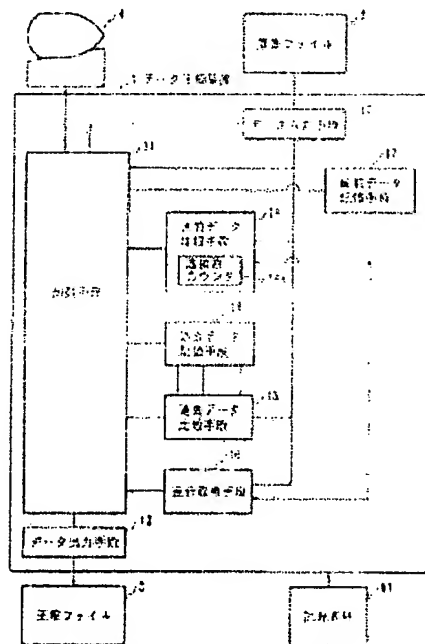
(72)Inventor : TAKIZAWA TETSUO

(54) METHOD FOR COMPRESSING DATA AND MACHINE READABLE RECORDING MEDIUM
STORING COMPRESSION PROGRAM

(57)Abstract:

PROBLEM TO BE SOLVED: To attain a high compression ratio independently of the kind of an image.

SOLUTION: In each input of a pixel value from an image to be compressed, a comparing means 14 compares the pixel value with a preceding pixel value just before stored in a storage means 17, and when both the values match with each other, counts up the contents of a counter 14a and processes a succeeding pixel value. At the time of nonmatching, a matching code setting up a continuous number is outputted when the value of the counter 14a is ≥ 1 , and then a comparing means 15 checks whether the pixel value matches any one of past pixel values stored in a storage means 18 or not. When matching is obtained, an index code setting up the index number of the pixel value is outputted, the contents of the storage means 17 are rewritten and a succeeding pixel value is processed. When there is no matching pixel value in the past pixel values, a difference acquiring means 16 finds out a difference from the just preceding pixel value, and when the difference value is less than a prescribed value, outputs a difference code setting up the difference value. In the other cases, the means 16 outputs an immediate value code setting up the pixel itself, the contents of the storage means 17, 18 are rewritten and a succeeding pixel value is processed.



*** NOTICES ***

JPO and INPIT are not responsible for any damages caused by the use of this translation.

1.This document has been translated by computer. So the translation may not reflect the original precisely.

2.**** shows the word which can not be translated.

3.In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1]Coincidence numerals which set up the number of times which the same data value as the last data value follows, A differential code which set up difference with the last data value, and index numerals which set up an index number of a processed data value memorized by past data memory measure, A data compression method using a total of four kinds of numerals with immediate numerals which set up the data value itself, choosing numerals whose data volume after compression decreases more for every data sequentially from a head of a data constellation used as a compression object, and compressing data.

[Claim 2]The data compression method according to claim 1 making numerals of all the kinds into numerals of a byte unit.

[Claim 3]The data compression method according to claim 1 or 2 using a differential code which is two or more kinds from which the greatest difference value that can be set up differs as a differential code.

[Claim 4]A data compression method which compresses data by the following steps.

(a) A pixel value inputted at Step (c) step b which inputs one pixel value from a step (b) compression object image which initializes a just before data storage means, a past data memory measure, and a consecutive number counter, and a pixel value memorized by just before data storage means. ***** a consecutive number counter at Step (d) step c to compare at the time of comparison coincidence, and. If a consecutive number counter has reached the maximum, after outputting coincidence numerals which set up the number of times of continuation and initializing a consecutive number counter, at Step (e) step c which returns to Step b At the time of comparison disagreement. If a consecutive number counter is except an initial value, after outputting coincidence numerals which set up the number of times of continuation and initializing a consecutive number counter, When an index number is called for at Step (f) step e which asks for an index number of a pixel value which searched a past data memory measure which attaches an index number and holds a pixel value which appeared in the past, and was inputted at Step b, and a pixel value in agreement, When index numerals which set up an index number are outputted and an index number cannot be found at Step (g) step e which stores in a just before data storage means a pixel value inputted at Step b, and returns to Step b, When difference for which a differential code of size smaller than size of immediate numerals which set up a pixel value inputted at Step (h) step b which asks for difference of a pixel value inputted at Step b and a pixel value memorized by just before data storage means was asked at Step g can be set up, Output a differential code which set up difference called for at Step g, and immediate numerals which set up a pixel value inputted at Step b when that was not right are outputted, A step which a pixel value inputted at Step b is stored in a just before data storage means, and stores also in a past data memory measure with a predetermined storing method, and returns to Step b [Claim 5](i) Whenever it finishes processing of a pixel number defined beforehand, if a consecutive number counter is except an initial value, will output coincidence numerals which set up the number of times of continuation, and a just before data storage means, a past data memory measure, and a consecutive number counter are initialized, The data compression method according to claim 4 having further a step which returns to Step b.

[Claim 6]The data compression method according to claim 4 or 5 making said each numerals into numerals of a byte unit.

[Claim 7]The data compression method according to claim 4, 5, or 6 using a differential code which is two or more kinds from which the greatest difference value that can be set up differs as a differential code.

[Claim 8]A recording medium which recorded a compression program which performs the following steps on a computer and in which machinery reading is possible.

(a) A pixel value inputted at Step (c) step b which inputs one pixel value from a step (b) compression object image which initializes a just before data storage means, a past data memory measure, and a consecutive number counter, and a pixel value memorized by just before data storage means. ***** a consecutive number counter at Step (d) step c to compare at the time of comparison coincidence, and. If a consecutive number counter has reached the maximum, after outputting coincidence numerals which set up the number of times of continuation and initializing a consecutive number counter, at Step (e) step c which returns to Step b At the time of comparison disagreement. If a consecutive number counter is except an initial value, after outputting coincidence numerals which set up the number of times of continuation and initializing a consecutive number counter, When an index number is called for at Step (f) step e which asks for an index number of a pixel value which searched a past data memory measure which attaches an index number and holds a pixel value which appeared in the past, and was inputted at Step b, and a pixel value in agreement, When index numerals which set up an index number are outputted and an index number cannot be found at Step (g) step e which stores in a just before data storage means a pixel value inputted at Step b, and returns to Step b, When difference for which a differential code of size smaller than size of immediate numerals which set up a pixel value inputted at Step (h) step b which asks for difference of a pixel value inputted at Step b and a pixel value memorized by just before data storage means was asked at Step g can be set up, Output a differential code which set up difference called for at Step g, and immediate numerals which set up a pixel value inputted at Step b when that was not right are outputted, A step which a pixel value inputted at Step b is stored in a just before data storage means, and stores also in a past data memory measure with a predetermined storing method, and returns to Step b [Claim 9]A recording medium which recorded on a computer the compression program according to claim 8 performing the following steps further and in which machinery reading is possible.

(i) A step which coincidence numerals which set up the number of times of continuation will be outputted if a consecutive number counter is except an initial value whenever it finishes processing of a pixel number defined beforehand, and initializes a just before data storage means, a past data memory measure, and a consecutive number counter, and returns to Step b

[Translation done.]

*** NOTICES ***

JPO and INPIT are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention]Especially this invention relates to the lossless compression method about a data compression method.

[0002]

[Description of the Prior Art]There are two kinds such as the lossy compression method and the lossless compression method of compression methods for a picture. Although the lossy compression method can generally realize a high compression ratio, since a certain distortion is added in process of compression and extension, it cannot reproduce the original picture thoroughly. On the other hand, in the compression ratio, since the lossless compression method can reproduce the original picture faithfully even if it passes through the process of compression and extension although it is generally inferior to the lossy compression method, the object for prizes of it is carried out in the field which is not allowed image quality deterioration. This invention relates to the lossless compression of this latter.

[0003]As the lossless compression method for a picture, the following methods are proposed thru/or put in practical use conventionally.

[0004](1) How to compress by Huffman encoding or arithmetic coding after reducing the entropy of information by taking the difference between pixels etc. so that it may be represented by the reversible mode (lossless mode) of finite difference method JPEG.

(2) How to compress the number (run length) which the pixel of the same value is following by recording with the pixel value so that it is represented by run length method packbits (one of the compression technology of a TIFF format).

(3) How to compress by searching from a dictionary the pixel row that whose it has existing registered [which carries out a longest match to the pixel row of an input] it has been coded so that it may be represented by the dictionary method LZW method, and coding with the reference number of a dictionary.

[0005]

[Problem(s) to be Solved by the Invention]As mentioned above, various kinds of methods are known from before as the lossless compression method for a picture, but each has merits and demerits. For example, to specific pictures, such as a binary format image which consists of a field that the processing of a run length method is simple simply and big, although a compression ratio is good, natural drawing like a photograph can hardly be compressed, but it becomes impossible compressing it. Although Argo RUZUMU is easy and is well used to file compression etc., a dictionary method cannot be compressed unless the same pattern appears repeatedly theoretically, but may become compression impossible. On the other hand, although a finite difference method is compressible uniformly regardless of the kind of picture in these three kinds of conventional methods, the compression ratio to the picture which the part, a run length method, and a dictionary method make elated is inferior to them. Since code length is strange good at bitwise, Huffman coding/algebraic sign also has the problem that data processing in bitwise is required and processing becomes complicated.

[0006]This invention is proposed in view of such a situation, and the 1st purpose is to provide

the compression method which serves as the highest compression ratio among the three above-mentioned kinds of conventional methods to various kinds of pictures, and the data compression method which can obtain the compression ratio which is not.

[0007]The purpose of this invention has data processing in bitwise in providing an unnecessary data compression method.

[0008]

[Means for Solving the Problem]In order to attain the 1st purpose of the above, this invention prepares two or more kinds of methods which decrease information entropy of a picture, and applies a method with the highest compression ratio from these methods for every pixel. That is, coincidence numerals which set up the number of times which the same data value as the last data value follows, A differential code which set up difference with the last data value, and index numerals which set up an index number of a processed data value memorized by past data memory measure, A total of four kinds of numerals with immediate numerals which set up the data value itself are used, and for every data, data volume after compression chooses numerals which decrease more, and is made to compress data sequentially from a head of a data constellation used as a compression object. When aimed at a picture, more specifically, data is compressed by the following steps.

[0009](a) A pixel value inputted at Step (c) step b which inputs one pixel value from a step (b) compression object image which initializes a just before data storage means, a past data memory measure, and a consecutive number counter, and a pixel value memorized by just before data storage means. ***** a consecutive number counter at Step (d) step c to compare at the time of comparison coincidence, and. If a consecutive number counter has reached the maximum, after outputting coincidence numerals which set up the number of times of continuation and initializing a consecutive number counter, at Step (e) step c which returns to Step b At the time of comparison disagreement. If a consecutive number counter is except an initial value, after outputting coincidence numerals which set up the number of times of continuation and initializing a consecutive number counter, When an index number is called for at Step (f) step e which asks for an index number of a pixel value which searched a past data memory measure which attaches an index number and holds a pixel value which appeared in the past, and was inputted at Step b, and a pixel value in agreement, When index numerals which set up an index number are outputted and an index number cannot be found at Step (g) step e which stores in a just before data storage means a pixel value inputted at Step b, and returns to Step b, When difference for which a differential code of size smaller than size of immediate numerals which set up a pixel value inputted at Step (h) step b which asks for difference of a pixel value inputted at Step b and a pixel value memorized by just before data storage means was asked at Step g can be set up, Output a differential code which set up difference called for at Step g, and immediate numerals which set up a pixel value inputted at Step b when that was not right are outputted, A step which a pixel value inputted at Step b is stored in a just before data storage means, and stores also in a past data memory measure with a predetermined storing method, and returns to Step b [0010]This invention makes numerals of all the kinds numerals of a byte unit, in order to also attain the 2nd purpose of the above.

[0011]Furthermore, a differential code which is two or more kinds from which the greatest difference value that can be set up differs is being used for this invention as a differential code.

[0012]In order for this invention to attain the 1st purpose of the above and to be able to perform an expansion process from the middle of compressed data at high speed, it has the following steps.

(i) A step which coincidence numerals which set up the number of times of continuation will be outputted if a consecutive number counter is except an initial value whenever it finishes processing of a pixel number defined beforehand, and initializes a just before data storage means, a past data memory measure, and a consecutive number counter, and returns to Step b [0013]

[Embodiment of the Invention]Next, the example of an embodiment of the invention is explained in detail with reference to Drawings.

[0014]Drawing 1 is a block diagram showing an example of the data compression device which applied this invention. The data compression device 1 of this example has the control means 11,

the data input means 12, the data output means 13, the just before data comparison means 14, the past data comparison means 15, and a function means of the difference acquisition means 16, and. It has memory measures, such as the just before data storage means 17 and the past data memory measure 18. It is connected to the graphics file 2, the compressed file 3, and the user input/output device 4.

[0015]The data input means 12 is a means to input the image data made into a compression object from the graphics file 2. At this example, the pixel value of image data is respectively expressed with the three color values R, G, and B of 8 bits. When it performs rightward from the pixel at the left end of a picture, the scanning order foreword, i.e., an entry-of-data order, of a picture, and it becomes a right end pixel, it is what is called a raster method again held rightward from the left end of the next line. The input origin of the image data made into a compression object is not restricted to a file.

[0016]The just before data storage means 17 is a means to hold the pixel value in front of the pixel value made into the present processing object.

[0017]The past data memory measure 18 is a means to memorize two or more kinds of pixel values which appeared in the past. In this example, as shown in drawing 2, the pixel value of a maximum of eight pieces is recorded. In drawing 2, E0-E7 are entries, and the index numbers 0-7 are attached, respectively. The column of the pixel value of each entries E0-E7 is made into the oar 0 at the time of initialization, whenever the pixel value which should be memorized appears, it is used sequentially from the young entry of an index number, and a pixel value is remembered to be entry E0 → E1 → —. Where a pixel value is memorized to the entry E7 of the index number 7, when memorizing a new pixel value, it returns to the entry E0 of the index number 0, and reuses sequentially from the young entry of an index number again. That is, it overwrites.

[0018]The just before data comparison means 14 is a means to compare a pixel value just before the pixel value inputted by the data input means 12 this time and the just before data storage means 17 memorize. When both are in agreement, it *****s the internal consecutive number counter 14a only 1. In the case of this example, the consecutive number counter 14a is countable from 0 to a maximum of 32 of an initial value.

[0019]The past data comparison means 15 is a means to compare the pixel value which appeared in the past memorized by the pixel value inputted by the data input means 12 this time, and the past data memory measure 18. When a pixel value in agreement exists in the past data memory measure 18, the index number is taken out.

[0020]The difference acquisition means 16 is a means to ask for the difference for every color value with a pixel value just before the pixel value inputted by the data input means 12 this time and the just before data storage means 17 memorize.

[0021]The data output means 13 is a means to output compressed data to the compressed file 3. The output destination change of compressed data is not restricted to a file.

[0022]The control means 11 is a means to control the data compression device 1 whole, and based on the processing result of the just before data comparison means 14, the past data comparison means 15, and the difference acquisition means 16, Sequentially from the pixel of the head of the picture used as a compression object, for every pixel, the data volume after compression chooses the numerals which decrease more, compresses data, and outputs compressed data to the compressed file 3 through the data output means 13.

[0023]Drawing 3 is an example of a format of the compression code used by this example. As shown in the figure, in this example, a total of six kinds of compression codes of 1 byte of coincidence numerals (the figure (a)), 1 byte of index numerals (the figure (b)), 1 byte, 2 bytes, and 3 bytes of differential code (the figure (c) - (e)) and 4 bytes of immediate numerals (the figure (f)) are used. All serve as data of the byte unit so that data processing in bitwise may become unnecessary.

[0024]The immediate numerals of drawing 3 (f) are used when outputting the incompressible pixel value itself, such as a head pixel of a picture. Each color value of R, G, and B goes into 3 bytes which the fixed bit string (11111000) the 1st byte indicates it to be that the numerals concerned are immediate numerals enters and follows.

[0025]The coincidence numerals of drawing 3 (a) are used when compressing the portion which

the same pixel value follows. The fixed bit string (110) the 5th bit – the 7th bit indicate it to be that the numerals concerned are coincidence numerals enters, and a length paragraph goes into 4th bit [the 0th bit –] a total of 5 bits. A length paragraph shows the how many same back pixel values as the last pixel value continue, and it is shown for the two same pixels as a just before pixel continuing that the one same pixel as a just before pixel continues at the time of 0 at the time of 1, respectively -- and that the 32 same pixels as a just before pixel continue at the time of the maximum 31.

[0026]The index numerals of drawing 3 (b) are used when compressing the same pixel value as the pixel value memorized by the past data memory measure 18. The fixed bit string (11110) it is triplet eye – 7 bit indicated to be that the numerals concerned are index numerals enters, and an index paragraph goes into 2nd bit [the 0th bit –] a total of 3 bits. The index number of the pixel value which was the same value in the past data memory measure 18 is set to an index paragraph.

[0027]Drawing 3 (c) The differential code of – (e) is used when compressing a pixel value by difference with the last pixel value, and as for the 1-byte differential code of 1 byte format, and (d), the 2-byte differential code of 2 byte format and (e) of (c) are the 3-byte differential codes of 3 byte format.

[0028]The fixed bit (0) which shows that the numerals concerned are 1-byte differential codes to the 7th bit with a 1-byte differential code enters, Difference deltaR of the color value R goes into 6th bit a total of 2 bits with the 5th bit, difference deltaG of the color value G goes into a triplet eye and 4th bit a total of 2 bits, and difference deltaB of the color value B goes into the triplet (the 0th bit – 2nd bit), respectively.

[0029]The fixed bit string (10) which shows that the numerals concerned are 2-byte differential codes to 7th bit [1st byte the 6th bit and] a total of 2 bits with a 2-byte differential code enters, Difference deltaG goes into 7th bit [the 5th bit –] a total of 4 bits of the byte to whom difference deltaR follows 1st byte 5th bit [the 1st bit –] a total of 5 bits with 1st byte the 0th bit, and difference deltaB goes into 2nd byte 4th bit [the 0th bit –] a total of 5 bits, respectively.

[0030]The fixed bit string (1110) which shows that the numerals concerned are 3-byte differential codes to 1st byte 7th bit [the 4th bit –] a total of 4 bits with a 3-byte differential code enters, To 7th bit [the 5th bit –] a total of 7 bits of the 1st byte of the 0th bit – a triplet eye, and the continuing byte, difference deltaR, Difference deltaG goes into 7th bit a total of 6 bits of the continuing byte with the 0th bit – 2nd byte the 4th bit, and difference deltaB goes into 3rd byte 6th bit [the 0th bit –] a total of 7 bits, respectively.

[0031]R with the last pixel value, G, and B -- a 1-byte differential code being used when each difference delta R, delta G, and delta B can express by 2, 2, and below a triplet, respectively, and, If it can express by 5 and 4 or 5 bits or less, a 2-byte differential code will be used, and if it can express by 7 and 6 or 7 bits or less, a 3-byte differential code will be used.

[0032]Drawing 4 is a flow chart which shows the example of processing of the data compression device 1 of drawing 1. Hereafter, operation of this example is explained with reference to drawing 1 thru/or drawing 4. When the computer which has CPU and a memory realizes the data compression device 1, The compression program which described the processing shown in drawing 4 is recorded on the recording medium (M1 of drawing 1) which can machinery read CD-ROM, semiconductor memory, a magnetic disk, etc., When the compression program of the recording medium M1 is read by the computer and controls operation of a computer, the computer is operated as the data compression device 1 of drawing 1.

[0033]If the compression demand which specified the graphics file 2 which stores the image data used as a compression object, and the compressed file 3 which stores a compressed image is inputted into the data compression device 1 from the user input/output device 4, the processing shown in drawing 4 will be started under control of the control means 11 of the data compression device 1.

[0034]The control means 11 initializes each part of a device first (S1). By this processing, all the pixel values in the just before data storage means 17 and the past data memory measure 18 are initialized by 0, and the consecutive number counter 14a within the just before data comparison

means 14 is also initialized by 0.

[0035]Next, the control means 11 inputs the first pixel value from the graphics file 2 using the data input means 12 (S2). The inputted pixel value is transmitted to the control means 11, the just before data comparison means 14, the past data comparison means 15, and the difference acquisition means 16.

[0036]Since the control means 11 is a pixel value of the beginning after initialization (it is YES at S4), The immediate numerals which set up the transmitted pixel value are outputted to the data output means 13, and (S5) it stores in the just before data storage means 17 (S6), and also stores in the entry E0 of the index number 0 of the past data memory measure 18 (S7). And it returns to Step S2 and the following pixel value is inputted. When the data output means 13 is accumulated each time or in fixed quantity, it bundles up the numerals outputted from the control means 11, and outputs them to the compressed file 3.

[0037]About the pixel value of the 2nd henceforth, it may be compressible using the last pixel value or the past pixel value.

[0038]First, the control means 11 reads in the just before data comparison means 14 a comparison result with a pixel value just before the pixel value inputted this time and the just before data storage means 17 memorize, and judges whether both are in agreement (S8). When both are in agreement, only 1 *****s the consecutive number counter 14a (S9). And it investigates whether the counted value of the consecutive number counter 14a after increment reached the maximum (this example 32) (S10), and if it has not reached, it returns to Step S2 and the following pixel value is processed. When having reached the maximum and the 32 same pixel values as a pixel value just before it got it blocked and the just before data storage means 17 memorized appear continuously, The coincidence numerals which set the value (31) of counted value-1 as the length paragraph of drawing 3 (a) are outputted to the data output means 13, the value of the consecutive number counter 14a is initialized to 0 (S11), it returns to Step S2, and the following pixel value is processed.

[0039]On the other hand, when a pixel value just before the pixel value inputted this time and the just before data storage means 17 memorize is not in agreement (it is NO at S8), Although it investigates whether compression by other compression format is possible for the control means 11 (S14, S15), Before that, it investigates whether the value of the consecutive number counter 14a is larger than zero (S12), and in being larger than 0, the coincidence numerals which set the value of counted value-1 as the length paragraph of drawing 3 (a) are outputted to the data output means 13, and it initializes the value of the consecutive number counter 14a 0 (S13).

[0040]Now, in Step S14, a comparison result with the pixel value of the past memorized by the pixel value inputted this time and the past data memory measure 18 is read in the past data comparison means 15, and the existence of the past data which is in agreement with the pixel value inputted this time is judged. Since the past data comparison means 15 outputs the index number of the entry which stores the congruous pixel values when past data in agreement exists, the control means 11, The index numerals which set this index number as the Index paragraph of drawing 3 (b) are outputted to the data output means 13. And this pixel value is stored in the just before data storage means 17 (S17), it returns to Step S2, and the following pixel value is processed.

[0041]When the same pixel value does not exist in the pixel value which appeared in the past (it is NO at S14), it is investigated whether compression by a differential code is possible (S15). The difference delta R, delta G, and delta B for every color value with a pixel value just before this pixel value and the just before data storage means 17 which were searched for in the difference acquisition means 16 memorize the control means 11, When it can express by 2, 2, and below a triplet, respectively, it judges that compression by a 1-byte differential code is possible, and the 1-byte differential code of drawing 3 (c) which set up the difference delta R, delta G, and delta B is outputted to the data output means 13 (S18). When the difference delta R, delta G, and delta B can express by 5 and 4 or 5 bits or less, respectively, it judges that compression by a 2-byte differential code is possible, and it outputs the 2-byte differential code of drawing 3 (d) which set up the difference delta R, delta G, and delta B to the data output means 13 (S18). When the difference delta R, delta G, and delta B can express by 7 and 6 or 7 bits or less, respectively, it

judges that compression by a 3-byte differential code is possible, and it outputs the 3-byte differential code of drawing 3 (e) which set up the difference delta R, delta G, and delta B to the data output means 13 (S18). When other (i.e., when each of the difference delta R, delta G, and delta B cannot express by 7 and 6 or 7 bits or less), it judges that the compression by a differential code is impossible, and the immediate numerals of drawing 3 (f) which set up this pixel value itself are outputted to the data output means 13 (S19).

[0042] And when a differential code and immediate numerals are outputted as mentioned above, this pixel value is stored in the just before data storage means 17, and (S6) it stores in the entry of the next index number of the past data memory measure 18 (S7). And it returns to Step S2 and the following pixel value is inputted.

[0043] After the above processings are repeated and finishing processing to the pixel of the last in the graphics file 2 (it is YES at S3), if the value of the consecutive number counter 14a is larger than zero, the coincidence numerals which set the value of counter value-1 as the length paragraph of drawing 3 (a) will be outputted (S20, S21), and compression processing will be ended.

[0044] The example of measurement of the compression ratio at the time of actually compressing into drawing 5 using the data compression method and the conventional method by this example about some pictures is shown. As the conventional method, packbits, the LZW method, and JPEG (lossless mode) were used. The numerical value in a figure shows "data volume of data volume/[after compression] origin", and shows that a compression ratio is so high that a numerical value is small. 1.00 shows compression impossible. The picture 1 is a picture on the computer screen where imaged figures where the character string was displayed, such as two windows and calculators, are displayed here, The picture 2 is a picture of a certain homepage screen displayed on the computer screen, The picture 3 is a picture in a certain PhotoDisc from which the pocket watch was placed on the opened book where a lid is opened, the picture 4 is the natural drawing which photoed the scenery that a lake is in this side and a forest is in the back, and the picture 5 is a pattern image in which many muscle-like patterns were drawn. receiving various kinds of pictures according to the data compression method of this example, as shown in this drawing 5 — the 1st — or it is compressible into the 2nd with a high compression ratio.

[0045] In the above working example, as application priority order of a compression code, top priority was given to coincidence numerals, then, priority was given to index numerals, and the differential code was applied at the end. This is because 1 byte and index numerals are 1 byte, coincidence numerals is [a differential code] a maximum of 3 bytes in the case of this example, so this application priority order is an order which can lessen the amount of information more. However, the number of bytes of each numerals is not limited to an above-mentioned example, but can adopt various kinds of things, and may also change the application priority order of a compression code in that case. For example, when using 2 bytes of coincidence numerals which made the length paragraph 13 bits, the same pixel value in the part which continues only two pieces. Since the amount of information has less it if expression by index numerals or a 1-byte differential code is possible, index numerals or a 1-byte differential code is preferentially applied from coincidence numerals.

[0046] Next, the picture compressed by the data compression method of above-mentioned working example is elongated, and working example of the data expanding method which restores the original picture is described.

[0047] Drawing 6 is a block diagram showing an example of the data extending apparatus which applied this invention. The data extending apparatus 5 of this example The control means 51 and the data input means 52, It has the data output means 53, the continuation pixel output means 54, the past data output means 55, the difference data output means 56, and the function means said selector 59-1, 59-2, and has memory measures, such as the just before data storage means 57 and the past data memory measure 58. It is connected to the compressed file 3, the restoration file 6, and the user input/output device 7.

[0048] The data input means 52 is a means to input the compressed data made applicable to restoration from the compressed file 3. As mentioned above, compressed data comprises a total

of six kinds of compression codes of 1 byte of coincidence numerals (drawing 3 (a)), 1 byte of index numerals (drawing 3 (b)), 1 byte, 2 bytes, and 3 bytes of differential code (drawing 3 (c) - (e)), and 4 bytes of immediate numerals (drawing 3 (f)). The data input means 52 inputs these compression codes sequentially from the head of compressed data. It can be distinguished in the identifier bit of numerals what byte numerals a pause of numerals is a byte unit and they are. That is, if read 1 byte the 7th bit is 0, it is a 1-byte differential code, If it is a 2-byte differential code if the 7 or 6th bit is 1 and 0, and 7 and the 6 or 5th bit are 1, 1, and 0, they are 1 byte of coincidence numerals, They are immediate numerals if it is index numerals if it is a 3-byte differential code if 7, 6, and the 5 or 4th bit are 1, 1, 1, and 0, and 7, 6, 5, 4, and a triplet eye are 1, 1, 1, 1, and 0, and 7, 6, 5, 4, 3, 2, 1, and 0 are 1, 1, 1, 1, 1, 0, 0, and 0. The input origin of the compressed data made applicable to restoration is not restricted to a file.

[0049]The just before data storage means 57 is a means to hold the pixel value restored immediately before.

[0050]The past data memory measure 58 is a means to memorize two or more kinds of pixel values restored in the past, and the storing method of the structure and the pixel value to that of it is the same as that of the past data memory measure 18 by the side of a data compression device. That is, in the case of this example, as shown in drawing 2, the pixel value of a maximum of eight pieces is memorized in the same order as a data compression device.

[0051]The continuation pixel output means 54 is an extension means of coincidence numerals, and only the value+1 of the length paragraph of the coincidence numerals into which the pixel value memorized by the just before data storage means 57 was inputted by the data input means 52 outputs it continuously.

[0052]The past data output means 55 is an extension means of index numerals, and reads and outputs the pixel value memorized by the past data memory measure 58 corresponding to the index number which the index numerals inputted by the data input means 52 show.

[0053]The difference data output means 56 is an extension means of a differential code, adds delta R, delta G, and delta B in the differential code inputted by the data input means 52 to each color value R, G, and B of the pixel value memorized by the just before data storage means 57, and restores and outputs a pixel value.

[0054]Based on the control signal from the control means 51 for any of the pixel value outputted from the continuation pixel output means 54, the difference data output means 56, and the control means 51 they are, choose the selector 59-1, and output to the selector 59-2, and. Since it stores as past data, it outputs to the past data memory measure 58.

[0055]Based on the control signal from the control means 51 for any of the pixel value outputted from the selector 59-1 and the past data output means 55 they are, choose the selector 59-2, and it is outputted to the data output means 53, and since it stores as a last pixel value, it is outputted to the just before data storage means 57.

[0056]The data output means 53 is a means to output the restored pixel value to the restoration file 6. The output destination change of restored data is not restricted to a file.

[0057]The control means 51 is a means to control the data extending apparatus 5 whole.

[0058]Drawing 7 is a flow chart which shows the example of processing of the data extending apparatus 5 of drawing 6. Hereafter, operation of this example is explained with reference to drawing 6 and drawing 7. When the computer which has CPU and a memory realizes the data extending apparatus 5, The extension program which described the processing shown in drawing 7 is recorded on the recording medium (M2 of drawing 6) which can machinery read CD-ROM, semiconductor memory, a magnetic disk, etc., When the extension program of the recording medium M2 is read by the computer and controls operation of a computer, the computer is operated as the data extending apparatus 5 of drawing 6.

[0059]If the extension demand which specified the compressed file 3 which stores the compressed data used as the candidate for restoration, and the restoration file 6 which stores image restoration is inputted into the data extending apparatus 5 from the user input/output device 7, the processing shown in drawing 7 will be started under control of the control means 51 of the data extending apparatus 5.

[0060]The control means 51 initializes each part of a device first (S31). By this processing, all

the pixel values in the just before data storage means 57 and the past data memory measure 58 are initialized by 0.

[0061]Next, the control means 51 inputs the first compression code from the compressed file 3 using the data input means 52 (S32), and judges the kind of the numerals (S34-S36).

[0062]When compression codes are immediate numerals (it is NO at S36), the pixel value itself set as immediate numerals is sent to the data output means 53 through selector 59-1, 59-2 from the control means 51, and it outputs to the restoration file 6 (S37). And it memorizes to the just before data storage means 57 by making the pixel value into a just before pixel value, and (S38) it memorizes as one of the past data to the past data memory measure 58 (S39), it returns to Step S32, and the following compression code is inputted.

[0063]When compression codes are coincidence numerals (it is YES at S34), the control means 51 transmits the coincidence numerals to the continuation pixel output means 54, The continuation pixel output means 54 the pixel value memorized by the just before data storage means 57, Only the number of the value+1 of the length paragraph of the coincidence numerals is continuously outputted to the selector 59-1, and the control means 51 makes the output choose by selector 59-1, 59-2, and is outputted to the restoration file 6 through the data output means 53 (S40). And it memorizes to the just before data storage means 57 by making the pixel value into a just before pixel value, and (S38) it memorizes as one of the past data to the past data memory measure 58 (S39), it returns to Step S32, and the following compression code is inputted.

[0064]When compression codes are index numerals (it is YES at S35), the control means 51 transmits the index numerals to the past data output means 55, The past data output means 55 reads the pixel value memorized by the past data memory measure 58 corresponding to the index number set as the index numerals, and outputs it to the selector 59-2, The control means 51 makes the output choose by the selector 59-2, and is outputted to the restoration file 6 through the data output means 53 (S41). And it memorizes to the just before data storage means 57 by making the pixel value into a just before pixel value (S42), it returns to Step S32, and the following compression code is inputted.

[0065]When a compression code is a differential code (it is YES at S36), the control means 51 transmits the differential code to the difference data output means 56, The difference data output means 56 to R of the pixel value memorized by the just before data storage means 57, G, and B value. The pixel value adding delta R, delta G, and delta B of the differential code is outputted to the selector 59-1, and the control means 51 makes the output choose by selector 59-1, 59-2, and is outputted to the restoration file 6 through the data output means 53 (S43). And it memorizes to the just before data storage means 57 by making the pixel value into a just before pixel value, and (S38) it memorizes as one of the past data to the past data memory measure 58 (S39), it returns to Step S32, and the following compression code is inputted.

[0066]An expansion process is ended, after the above processings are repeated and finishing processing to the compression code of the last in the compressed file 3 (it is YES at S33).

[0067]The block diagram showing another example of the data compression device with which drawing 8 applied this invention, and drawing 9 are flow charts which show the example of processing. The place where the data compression device 1 of this example is different from the data compression device of drawing 1, Whenever it finishes processing the pixel value for one line of a compression object image, it is in the point of having enabled it to perform the expansion process from the middle of compressed data at high speed so that it may mention later at the time of extension of the compressed file 3, by initializing a last pixel value, past data, etc. For this reason, in the control means 11, the pixel number counter 11a which counts the pixel number processed till the present is formed, and change of Steps S22-S26 being inserted between Step S1 and Step S2 is added. When the computer which has CPU and a memory realizes the data compression device 1, The compression program which described the processing shown in drawing 9 is recorded on the recording medium (M1 of drawing 8) which can machinery read CD-ROM, semiconductor memory, a magnetic disk, etc., When the compression program of the recording medium M1 is read by the computer and controls operation of a computer, the computer is operated as the data compression device 1 of drawing 8.

[0068] Hereafter, operation of this example is explained focusing on a difference with the data compression device of working example of drawing 1.

[0069] If the compression demand which specified the graphics file 2 which stores the image data used as a compression object, the compressed file 3 which stores a compressed image, and the pixel number L per line of a picture is inputted into the data compression device 1 from the user input/output device 4, The processing shown in drawing 9 is started under control of the control means 11 of the data compression device 1. The pixel number L scans the picture in the graphics file 3, and it may be made to acquire it automatically.

[0070] The control means 11 initializes each part of a device first (S1). In this processing, all the pixel values in the just before data storage means 17 and the past data memory measure 18 are initialized by 0, and the consecutive number counter 14a within the just before data comparison means 14 is also initialized by 0, and the pixel number counter 11a is initialized by 0.

[0071] Next, although the control means 11 inputs a pixel value one by one from the graphics file 2 using the data input means 12 and repeats the same processing as the data compression device of drawing 1 (S2-S21), in this example. Before processing one pixel value, the number of the pixels which finished processing the pixel number counter 11a by the present after (S22) and initialization because only 1 *****s is counted. And since processing for one line is finished when the value of the pixel number counter 11a amounts to L+1 (it is YES at S23), If the value of the consecutive number counter 14a is larger than zero, after performing processing which outputs the coincidence numerals which set the value of counted value-1 as the length paragraph of drawing 3 (a) to the data output means 13 (S24, S25), If the value of the consecutive number counter 14a is 0, the processing S25 will be skipped. It progresses to Step S26, respectively, all the pixel values in the just before data storage means 17 and the past data memory measure 18 are initialized to 0, and the consecutive number counter 14a within the just before data comparison means 14 is initialized to 0, and the pixel number counter 11a is set as 1. And it progresses to Step S2 and the following pixel value is processed.

[0072] Thus, whenever it finishes processing the pixel value for one line of a picture, some compression ratios will deteriorate by initializing again, but the expansion process from the middle of compressed data can be performed at high speed. Hereafter, the processing at the time of extension is explained.

[0073] Drawing 10 is a block diagram showing an example of the data extending apparatus which elongates the compressed image in the graphics file 3 generated with the data compression device of drawing 8. When the extension demand from the user input/output device 7 and the line which wants to start restoration are specified, the place where the data extending apparatus 5 of this example is different from the data extending apparatus of drawing 6 pulls out the head of the line, and is at the point of having a function which actually reverts from the pixel value after the line. For this reason, it is changed, as the pixel counter 51a, the line counter 51b, and the search means 51 are formed in the control means 11 and processing of a data extending apparatus is also shown in drawing 11 and drawing 12. When the computer which has CPU and a memory realizes the data extending apparatus 5, The extension program which described the processing shown in drawing 11 and drawing 12 is recorded on the recording medium (M2 of drawing 10) which can machinery read CD-ROM, semiconductor memory, a magnetic disk, etc., When the extension program of the recording medium M2 is read by the computer and controls operation of a computer, the computer is operated as the data extending apparatus 5 of drawing 10.

[0074] Hereafter, operation of this example is explained focusing on a difference with the data extending apparatus of working example of drawing 6.

[0075] If the extension demand which specified restoration start line number m the compressed file 3 which stores the compressed data used as the candidate for restoration, the restoration file 6 which stores image restoration, the pixel number L per line, and if needed is inputted into the data extending apparatus 5 from the user input/output device 7, The processing shown in drawing 11 is started under control of the control means 51 of the data extending apparatus 5. The pixel number per line is recorded into the compressed file 3 at the time of compression, and it may be made to read this. Restoration start line number m specifies a head line as No. 1. For

example, in the case of the 3rd line, it is referred to as $m=3$.

[0076]The control means 51 initializes each part of a device first (S51). By this processing, all the pixel values in the just before data storage means 57 and the past data memory measure 58 are initialized by 0, and the pixel number counter 51a and the line counter 51b are also initialized by 0.

[0077]Next, if it progresses to processing of drawing 12 and there is no such specification after it will perform processing of Steps S53-S61 by the search means 51c, if the control means 51 judges whether there was any specification of a restoration start line number (S52) and has such specification, it will progress to processing of drawing 12 promptly.

[0078]It reads one numerals of search processing by the search means 51c at a time sequentially from the compression code of the head of the compressed file 3 by the data input means 52 (S53). If the compression code is coincidence numerals (it is YES at S55), only the number of times of the value+1 of the length paragraph of the coincidence numerals will ***** the pixel number counter 51a (S56), and if it is the other compression code, only 1 ***** the pixel number counter 51a (S57). And when the counted value of the pixel number counter 51a after increment reaches the pixel number $L-1$ which is one line, if only 1 ***** the line counter 51b (S58, S59) and the counted value of the line counter 51b reaches the restoration start line number $m-1$ (it is YES at S60), it will become the completion of search. Only by calculating the pixel number which distinguishes the kind of numerals and is restored, since restoration of a actual pixel value is not performed, search can be performed at high speed. Before being judged as YES at Step S60, when processed to the last compression code, it means that specification of YES) and a restoration start line number had had an error by (S54, and an error message is outputted to the user input/output device 7 (S61).

[0079]Processing of drawing 12 is almost the same as that of drawing 7. However, since a last pixel value and past data are initialized for every line at the time of compression, Also at the time of extension, a restoration pixel number is counted using the pixel counter 51a (S71, S73). If it finishes restoring to the pixel of the last of one line (it is YES at S72 and S74), will return to Step S31 and all the pixel values in the just before data storage means 57 and the past data memory measure 58 will be initialized to 0, and the pixel number counter 51a is initialized to 0.

[0080]Although restoration from the head of arbitrary lines was enabled in working example of drawing 8 and drawing 10, restoration from the arbitrary lines of arbitrary lines can also be enabled. In that case, after specifying a restoration initial line number in addition to a restoration start line number and pulling out the head on the specified restoration start line, It stops, although the pixel value of a just before [a restoration initial line] is memorized as a just before pixel value and past data, without outputting, and usual processing actually outputted to the restoration file 6 from the pixel of a restoration initial line is performed.

[0081]

[Effect of the Invention]According to this invention, the following effects can be acquired as explained above.

[0082]In order to use a total of four kinds of numerals, coincidence numerals, a differential code, index numerals, and immediate numerals, properly, to choose the numerals whose data volume after compression decreases more for every data sequentially from the head of the data constellation used as a compression object and to compress data, The compression method which serves as the highest compression ratio among the conventional methods to various kinds of pictures, and the compression ratio which is not can be obtained.

[0083]If it is in the composition which makes the numerals of all the kinds the numerals of a byte unit, data processing in bitwise becomes unnecessary and the part and a throughput decrease.

[0084]If it is in the composition which initializes a last pixel value, past data, etc. whenever it finishes processing of the pixel number defined beforehand, the expansion process from the middle of compressed data can be performed at high speed at the time of extension.

[Translation done.]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-203457

(43) 公開日 平成11年(1999) 7月30日

(51) Int. Cl. ⁶
G06T 1/00
H03M 7/30
H04N 1/41

識別記号

F I

G06F 15/66
H03M 7/30
H04N 1/41

J
Z
C
B

審査請求 有 請求項の数 9 F D (全17頁)

(21) 出願番号 特願平10-17935

(22) 出願日 平成10年(1998) 1月14日

(71) 出願人 000004237

日本電気株式会社
東京都港区芝五丁目7番1号

(72) 発明者 滝澤 哲郎

東京都港区芝五丁目7番1号 日本電気株
式会社内

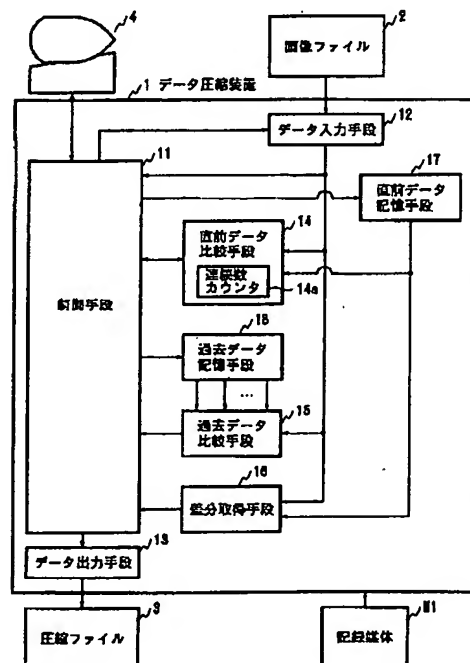
(74) 代理人 弁理士 境 廣巳

(54) 【発明の名称】 データ圧縮方法及び圧縮プログラムを記録した機械読み取り可能な記録媒体

(57) 【要約】

【課題】 画像の種類を問わずに高い圧縮率を達成する。

【解決手段】 圧縮対象画像から1つの画素値を入力する毎に、比較手段14により記憶手段17に記憶された直前の画素値と比較し、一致した場合はカウンタ14aをインクリメントし、次の画素値を処理する。一致しない場合、カウンタ14aの値が1以上であれば連続数を設定した一致符号を出力した後、比較手段15により記憶手段18に記憶された過去の何れかの画素値と一致するかを調べ、一致した場合にその画素値のインデックス番号を設定したインデックス符号を出力し、記憶手段17を書き換え、次の画素値を処理する。過去データに一致するものがない場合、差分取得手段16により直前の画素値との差分を求め、その差分値が所定値以下であればその差分値を設定した差分符号を出力し、それ以外は画素値そのものを設定した即値符号を出力し、記憶手段17及び記憶手段18を書き換え、次の画素値を処理する。



【特許請求の範囲】

【請求項 1】 直前のデータ値と同じデータ値が連続する回数を設定した一致符号と、直前のデータ値との差分を設定した差分符号と、過去データ記憶手段に記憶された処理済のデータ値のインデックス番号を設定したインデックス符号と、データ値そのものを設定した即値符号との合計 4 種類の符号を使用し、圧縮対象となるデータ群の先頭から順に 1 データ毎に、圧縮後のデータ量がより少なくなる符号を選択してデータの圧縮を行うようにしたことを特徴とするデータ圧縮方法。

【請求項 2】 全ての種類の符号をバイト単位の符号としたことを特徴とする請求項 1 記載のデータ圧縮方法。

【請求項 3】 差分符号として、設定できる最大の差分値が異なる複数種類の差分符号を使用することを特徴とする請求項 1 または 2 記載のデータ圧縮方法。

【請求項 4】 以下のステップによりデータを圧縮するデータ圧縮方法。

(a) 直前データ記憶手段、過去データ記憶手段および連続数カウンタを初期化するステップ

(b) 圧縮対象画像から 1 つの画素値を入力するステップ

(c) ステップ b で入力された画素値と直前データ記憶手段に記憶された画素値とを比較するステップ

(d) ステップ c で比較一致のとき連続数カウンタをインクリメントすると共に、連続数カウンタが最大値に達していれば連続回数を設定した一致符号を出力して連続数カウンタを初期化した後、ステップ b に戻るステップ

(e) ステップ c で比較不一致のとき、連続数カウンタが初期値以外であれば連続回数を設定した一致符号を出力して連続数カウンタを初期化した後、過去に出現した画素値をインデックス番号を付けて保持する過去データ記憶手段を検索してステップ b で入力された画素値と一致する画素値のインデックス番号を求めるステップ

(f) ステップ e でインデックス番号が求められたとき、インデックス番号を設定したインデックス符号を出力すると共に、ステップ b で入力された画素値を直前データ記憶手段に格納して、ステップ b に戻るステップ

(g) ステップ e でインデックス番号が求まらなかったとき、ステップ b で入力された画素値と直前データ記憶手段に記憶された画素値との差分を求めるステップ

(h) ステップ b で入力された画素値を設定した即値符号のサイズより小さなサイズの差分符号にステップ g で求められた差分を設定できるとき、ステップ g で求められた差分を設定した差分符号を出力し、そうでないときはステップ b で入力された画素値を設定した即値符号を出力し、ステップ b で入力された画素値を直前データ記憶手段に格納すると共に所定の格納方法で過去データ記憶手段にも格納し、ステップ b に戻るステップ

【請求項 5】 (i) 予め定められた画素数の処理を終える毎に、連続数カウンタが初期値以外であれば連続回

数を設定した一致符号を出力すると共に直前データ記憶手段、過去データ記憶手段および連続数カウンタを初期化して、ステップ b に戻るステップを更に有することを特徴とする請求項 4 記載のデータ圧縮方法。

【請求項 6】 前記各符号をバイト単位の符号としたことを特徴とする請求項 4 または 5 記載のデータ圧縮方法。

【請求項 7】 差分符号として、設定できる最大の差分値が異なる複数種類の差分符号を使用することを特徴とする請求項 4、5 または 6 記載のデータ圧縮方法。

【請求項 8】 コンピュータに以下のステップを実行させる圧縮プログラムを記録した機械読み取り可能な記録媒体。

(a) 直前データ記憶手段、過去データ記憶手段および連続数カウンタを初期化するステップ

(b) 圧縮対象画像から 1 つの画素値を入力するステップ

(c) ステップ b で入力された画素値と直前データ記憶手段に記憶された画素値とを比較するステップ

(d) ステップ c で比較一致のとき連続数カウンタをインクリメントすると共に、連続数カウンタが最大値に達していれば連続回数を設定した一致符号を出力して連続数カウンタを初期化した後、ステップ b に戻るステップ

(e) ステップ c で比較不一致のとき、連続数カウンタが初期値以外であれば連続回数を設定した一致符号を出力して連続数カウンタを初期化した後、過去に出現した画素値をインデックス番号を付けて保持する過去データ記憶手段を検索してステップ b で入力された画素値と一致する画素値のインデックス番号を求めるステップ

(f) ステップ e でインデックス番号が求められたとき、インデックス番号を設定したインデックス符号を出力すると共に、ステップ b で入力された画素値を直前データ記憶手段に格納して、ステップ b に戻るステップ

(g) ステップ e でインデックス番号が求まらなかったとき、ステップ b で入力された画素値と直前データ記憶手段に記憶された画素値との差分を求めるステップ

(h) ステップ b で入力された画素値を設定した即値符号のサイズより小さなサイズの差分符号にステップ g で求められた差分を設定できるとき、ステップ g で求められた差分を設定した差分符号を出力し、そうでないときはステップ b で入力された画素値を設定した即値符号を出力し、ステップ b で入力された画素値を直前データ記憶手段に格納すると共に所定の格納方法で過去データ記憶手段にも格納し、ステップ b に戻るステップ

【請求項 9】 コンピュータに更に以下のステップを実行させることを特徴とする請求項 8 記載の圧縮プログラムを記録した機械読み取り可能な記録媒体。

(i) 予め定められた画素数の処理を終える毎に、連続数カウンタが初期値以外であれば連続回数を設定した一致符号を出力すると共に直前データ記憶手段、過去データ

10

20

30

40

50

タ記憶手段および連続数カウンタを初期化して、ステップbに戻るステップ

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明はデータ圧縮方法に関し、特に可逆圧縮方法に関するものである。

【0002】

【従来の技術】画像を対象とした圧縮方法には、非可逆圧縮方法と可逆圧縮方法との2種類がある。非可逆圧縮方法は、一般に高い圧縮率を実現できるが、圧縮、伸張の過程でなんらかの歪みが付加されるため、完全には元の画像を再現することはできない。これに対し、可逆圧縮方法は圧縮率では一般に非可逆圧縮方法に劣るものの、圧縮、伸張の過程を経ても元の画像を忠実に再現することができるため、画質劣化が許されない分野で賞用されている。本発明は、この後者の可逆圧縮に関する。

【0003】画像を対象とした可逆圧縮方法としては、以下のような方法が従来より提案ないし実用化されている。

【0004】(1) 差分法

JPEGの可逆モード(ロスレスモード)に代表されるように、画素間の差分を取るなどにより情報のエントロピーを減じた後、ハフマン符号化や算術符号化により圧縮する方法。

(2) ランレングス法

packbits(TIFFフォーマットの圧縮方式の一つ)に代表されるように、同じ値の画素が連続している個数(ランレングス)をその画素値と共に記録することで圧縮する方法。

(3) 辞書法

LZW法に代表されるように、入力した画素列に最長一致する既登録の符号化済の画素列を辞書から検索して辞書の参照番号により符号化することで圧縮する方法。

【0005】

【発明が解決しようとする課題】上述したように画像を対象とした可逆圧縮方法として従来より各種の方法が知られているが、それぞれに一長一短がある。例えば、ランレングス法は処理が簡単であり、また単純で大きな領域からなる2値画像など特定の画像に対しては圧縮率が良いが、写真のような自然画はほとんど圧縮することができず、圧縮不能となる。また、辞書法はアルゴリズムが簡単でファイル圧縮等に良く使われているが、原理的に同じパターンが繰り返し現れないと圧縮できず、圧縮不能になる場合がある。これに対して差分法は、これら3種類の従来方法の中では画像の種類を問わず満遍なく圧縮することができるが、その分、ランレングス法や辞書法が得意とする画像に対する圧縮率はそれらより劣る。また、ハフマン符号/算術符号は符号長がビット単位で可変なため、ビット単位でのデータ処理が必要で処理が複雑になるという問題点もある。

【0006】本発明はこのような事情に鑑みて提案されたものであり、その第1の目的は、様々な種類の画像に対して上記3種類の従来方法のうち最も高い圧縮率となる圧縮方法と遜色ない圧縮率を得ることができるデータ圧縮方法を提供することにある。

【0007】また本発明の目的は、ビット単位でのデータ処理が不要なデータ圧縮方法を提供することにある。

【0008】

【課題を解決するための手段】本発明は上記第1の目的を達成するために、画像の情報エントロピーを減少させる方式を複数種類用意し、画素毎にこれらの方式から最も圧縮率の高い方式を適用する。つまり、直前のデータ値と同じデータ値が連続する回数を設定した一致符号と、直前のデータ値との差分を設定した差分符号と、過去データ記憶手段に記憶された処理済のデータ値のインデックス番号を設定したインデックス符号と、データ値そのものを設定した即値符号との合計4種類の符号を使用し、圧縮対象となるデータ群の先頭から順に1データ毎に、圧縮後のデータ量がより少なくなる符号を選択してデータの圧縮を行うようにしている。より具体的には、画像を対象とする場合、以下のステップによりデータを圧縮する。

【0009】(a) 直前データ記憶手段、過去データ記憶手段および連続数カウンタを初期化するステップ

(b) 圧縮対象画像から1つの画素値を入力するステップ

(c) ステップbで入力された画素値と直前データ記憶手段に記憶された画素値とを比較するステップ

(d) ステップcで比較一致のとき連続数カウンタをインクリメントすると共に、連続数カウンタが最大値に達していれば連続回数を設定した一致符号を出力して連続数カウンタを初期化した後、ステップbに戻るステップ

(e) ステップcで比較不一致のとき、連続数カウンタが初期値以外であれば連続回数を設定した一致符号を出力して連続数カウンタを初期化した後、過去に出現した画素値をインデックス番号を付けて保持する過去データ記憶手段を検索してステップbで入力された画素値と一致する画素値のインデックス番号を求めるステップ

(f) ステップeでインデックス番号が求められたとき、インデックス番号を設定したインデックス符号を出力すると共に、ステップbで入力された画素値を直前データ記憶手段に格納して、ステップbに戻るステップ

(g) ステップeでインデックス番号が求めなかったとき、ステップbで入力された画素値と直前データ記憶手段に記憶された画素値との差分を求めるステップ

(h) ステップbで入力された画素値を設定した即値符号のサイズより小さなサイズの差分符号にステップgで求められた差分を設定できるとき、ステップgで求められた差分を設定した差分符号を出力し、そうでないときはステップbで入力された画素値を設定した即値符号を

出力し、ステップbで入力された画素値を直前データ記憶手段に格納すると共に所定の格納方法で過去データ記憶手段にも格納し、ステップbに戻るステップ

【0010】また本発明は上記第2の目的をも達成するために、全ての種類の符号をバイト単位の符号としている。

【0011】さらに本発明は、差分符号として、設定できる最大の差分値が異なる複数種類の差分符号を使用している。

【0012】また本発明は、上記第1の目的を達成すると共に、圧縮データの途中からの伸張処理を高速に行えるようにするために、更に以下のステップを有する。

(i) 予め定められた画素数の処理を終える毎に、連続数カウンタが初期値以外であれば連続回数を設定した一致符号を出力すると共に直前データ記憶手段、過去データ記憶手段および連続数カウンタを初期化して、ステップbに戻るステップ

【0013】

【発明の実施の形態】次に本発明の実施の形態の例について図面を参照して詳細に説明する。

【0014】図1は本発明を適用したデータ圧縮装置の一例を示すブロック図である。この例のデータ圧縮装置1は、制御手段11と、データ入力手段12と、データ出力手段13と、直前データ比較手段14と、過去データ比較手段15と、差分取得手段16といった機能手段を有すると共に、直前データ記憶手段17と過去データ記憶手段18といった記憶手段を有している。また、画像ファイル2、圧縮ファイル3および利用者入出力装置4に接続されている。

【0015】データ入力手段12は、圧縮対象とする画像データを画像ファイル2から入力する手段である。本実施例では、画像データの画素値は各々8ビットの3つの色値R、G、Bで表されている。また、画像の走査順序、つまりデータの入力順序は、画像の左端の画素から右方向に行い、右端の画素に達したら次のラインの左端から再び右方向に行う所謂ラスタ方式である。なお、圧縮対象とする画像データの入力元はファイルに限られない。

【0016】直前データ記憶手段17は、現在処理対象としている画素値の直前の画素値を保持する手段である。

【0017】過去データ記憶手段18は、過去に出現した画素値を複数種類記憶する手段である。本実施例では、図2に示すように最大8個の画素値を記録する。図2において、E0～E7はエントリであり、それぞれインデックス番号0～7が付されている。初期化時、各エントリE0～E7の画素値の欄はオール0にされ、記憶すべき画素値が出現する毎に、エントリE0→E1→…と、インデックス番号の若いエントリから順に使用されて画素値が記憶される。インデックス番号7のエントリ

E7まで画素値を記憶した状態で新たな画素値を記憶する場合、インデックス番号0のエントリE0に戻り、再びインデックス番号の若いエントリから順に再利用する。つまり、上書きしていく。

【0018】直前データ比較手段14は、データ入力手段12で今回入力された画素値と直前データ記憶手段17に記憶されている直前の画素値とを比較する手段である。両者が一致した場合には内部の連続数カウンタ14aが1だけインクリメントされる。連続数カウンタ14aは、本実施例の場合、初期値の0から最大32までカウントできる。

【0019】過去データ比較手段15は、データ入力手段12で今回入力された画素値と過去データ記憶手段18に記憶されている過去に出現した画素値とを比較する手段である。一致する画素値が過去データ記憶手段18に存在する場合、そのインデックス番号が取り出される。

【0020】差分取得手段16は、データ入力手段12で今回入力された画素値と直前データ記憶手段17に記憶されている直前の画素値との、各色値ごとの差分を求める手段である。

【0021】データ出力手段13は、圧縮ファイル3に圧縮データを出力する手段である。なお、圧縮データの出力先はファイルに限られない。

【0022】制御手段11は、データ圧縮装置1全体を制御する手段であり、直前データ比較手段14、過去データ比較手段15および差分取得手段16の処理結果に基づき、圧縮対象となる画像の先頭の画素から順に1画素毎に、圧縮後のデータ量がより少なくなる符号を選択してデータの圧縮を行い、圧縮データをデータ出力手段13を通じて圧縮ファイル3に出力する。

【0023】図3は本実施例で使用される圧縮符号のフォーマット例である。同図に示すように、本実施例では、1バイトの一致符号(同図(a))、1バイトのインデックス符号(同図(b))、1バイト、2バイト及び3バイトの差分符号(同図(c)～(e))、4バイトの即値符号(同図(f))の合計6種類の圧縮符号を用いる。何れも、ビット単位でのデータ処理が不要になるようにバイト単位のデータとなっている。

【0024】図3(f)の即値符号は、画像の先頭画素など、非圧縮の画素値そのものを出力するときに使用される。1バイト目は当該符号が即値符号であることを示す固定のビット列(11111000)が入り、続く3バイトにR、G、Bの各色値が入る。

【0025】図3(a)の一致符号は、同じ画素値が連続する部分を圧縮する場合に使用される。5ビット目～7ビット目は当該符号が一致符号であることを示す固定のビット列(110)が入り、0ビット目～4ビット目の合計5ビットにレングス項が入る。レングス項は直前の画素値と同じ画素値が後何個連続するかを示し、0の

とき直前画素と同じ画素が1個続くことを、1のとき直前画素と同じ画素が2個続くことを、…、最大値31のとき直前画素と同じ画素が32個続くことを、それぞれ示す。

【0026】図3(b)のインデックス符号は、過去データ記憶手段18に記憶された画素値と同じ画素値を圧縮する場合に使用される。3ビット目～7ビット目は当該符号がインデックス符号であることを示す固定のビット列(11110)が入り、0ビット目～2ビット目の合計3ビットにインデックス項が入る。インデックス項には、過去データ記憶手段18中の同じ値であった画素値のインデックス番号が設定される。

【0027】図3(c)～(e)の差分符号は、直前の画素値との差分によって画素値を圧縮する場合に使用され、(c)は1バイト形式の1バイト差分符号、(d)は2バイト形式の2バイト差分符号、(e)は3バイト形式の3バイト差分符号である。

【0028】1バイト差分符号では、7ビット目に当該符号が1バイト差分符号であることを示す固定のビット(0)が入り、5ビット目と6ビット目の合計2ビットに色値Rの差分 ΔR が、3ビット目と4ビット目の合計2ビットに色値Gの差分 ΔG が、0ビット目～2ビット目の3ビットに色値Bの差分 ΔB が、それぞれ入る。

【0029】2バイト差分符号では、1バイト目の6ビット目と7ビット目の合計2ビットに当該符号が2バイト差分符号であることを示す固定のビット列(10)が入り、1バイト目の1ビット目～5ビット目の合計5ビットに差分 ΔR が、1バイト目の0ビット目と続くバイトの5ビット目～7ビット目の合計4ビットに差分 ΔG が、2バイト目の0ビット目～4ビット目の合計5ビットに差分 ΔB が、それぞれ入る。

【0030】3バイト差分符号では、1バイト目の4ビット目～7ビット目の合計4ビットに当該符号が3バイト差分符号であることを示す固定のビット列(1110)が入り、1バイト目の0ビット目～3ビット目と続くバイトの5ビット目～7ビット目の合計7ビットに差分 ΔR が、2バイト目の0ビット目～4ビット目と続くバイトの7ビット目の合計6ビットに差分 ΔG が、3バイト目の0ビット目～6ビット目の合計7ビットに差分 ΔB が、それぞれ入る。

【0031】直前の画素値とのR、G、Bそれぞれの差分 ΔR 、 ΔG 、 ΔB がそれぞれ2、2、3ビット以下で表現できる場合には1バイト差分符号が使用され、5、4、5ビット以下で表現できるならば2バイト差分符号が使用され、7、6、7ビット以下で表現できるならば3バイト差分符号が使用される。

【0032】図4は図1のデータ圧縮装置1の処理例を示すフローチャートである。以下、図1ないし図4を参照して本実施例の動作を説明する。なお、データ圧縮装置1をCPUおよびメモリを有するコンピュータで実現

する場合、図4に示される処理を記述した圧縮プログラムがCD-ROM、半導体メモリ、磁気ディスク等の機械読み取り可能な記録媒体(図1のM1)に記録されており、その記録媒体M1の圧縮プログラムがコンピュータに読み取られ、コンピュータの動作を制御することにより、そのコンピュータを図1のデータ圧縮装置1として機能させる。

【0033】圧縮対象となる画像データを格納する画像ファイル2、圧縮画像を格納する圧縮ファイル3を指定した圧縮要求が利用者入出力装置4からデータ圧縮装置1に入力されると、データ圧縮装置1の制御手段11の制御の下で、図4に示す処理が開始される。

【0034】まず制御手段11は、装置各部を初期化する(S1)。この処理で、直前データ記憶手段17及び過去データ記憶手段18中の画素値がすべて0に初期化され、直前データ比較手段14内の連続数カウンタ14aも0に初期化される。

【0035】次に制御手段11は、データ入力手段12を用いて画像ファイル2から最初の画素値を入力する(S2)。入力された画素値は、制御手段11、直前データ比較手段14、過去データ比較手段15、差分取得手段16に伝達される。

【0036】制御手段11は、初期化後の最初の画素値なので(S4でYES)、伝達された画素値を設定した即値符号をデータ出力手段13に出力すると共に(S5)、直前データ記憶手段17に格納し(S6)、更に過去データ記憶手段18のインデックス番号0のエントリE0に格納する(S7)。そして、ステップS2に戻って次の画素値を入力する。なお、データ出力手段13は制御手段11から出力された符号をその都度、あるいは一定量蓄積された時点で一括して圧縮ファイル3に出力する。

【0037】2番目以降の画素値については直前の画素値あるいは過去の画素値を利用して圧縮できる可能性がある。

【0038】先ず制御手段11は、今回入力された画素値と直前データ記憶手段17に記憶されている直前の画素値との比較結果を直前データ比較手段14から読み取り、両者が一致しているか否かを判定する(S8)。両者が一致していた場合、連続数カウンタ14aを1だけインクリメントする(S9)。そして、インクリメント後の連続数カウンタ14aのカウント値が最大値(本例では32)に達したか否かを調べ(S10)、達していなければステップS2に戻って次の画素値を処理する。また、最大値に達していれば、つまり、直前データ記憶手段17に記憶された直前の画素値と同じ画素値が32個連続して出現した場合には、図3(a)のレンジス項にカウント値-1の値(31)を設定した一致符号をデータ出力手段13に出力し、連続数カウンタ14aの値を0に初期化して(S11)、ステップS2に戻り、次

の画素値を処理する。

【0039】他方、今回入力された画素値と直前データ記憶手段17に記憶されている直前の画素値とが一致しなかった場合(S8でNO)、制御手段11は他の圧縮形式による圧縮が可能か否かを調べるが(S14, S15)、その前に、連続数カウンタ14aの値が0より大きいかな否かを調べ(S12)、0より大きい場合には、図3(a)のレンジス項にカウンタ値-1の値を設定した一致符号をデータ出力手段13に出力し、連続数カウンタ14aの値を0に初期化する(S13)。

【0040】さて、ステップS14では、今回入力された画素値と過去データ記憶手段18に記憶されている過去の画素値との比較結果を過去データ比較手段15から読み取り、今回入力された画素値と一致する過去データの有無を判定する。一致する過去データが存在していた場合、過去データ比較手段15はその一致した画素値を格納しているエントリのインデックス番号を出力するので、制御手段11は、このインデックス番号を図3

(b)のIndex項に設定したインデックス符号をデータ出力手段13に出力する。そして、今回の画素値を直前データ記憶手段17に格納し(S17)、ステップS2に戻って次の画素値を処理する。

【0041】また、過去に出現した画素値の中にも同一の画素値が存在しない場合(S14でNO)、差分符号による圧縮が可能か否かを調べる(S15)。制御手段11は、差分取得手段16において求められた、今回の画素値と直前データ記憶手段17に記憶されている直前の画素値との各色値ごとの差分 ΔR 、 ΔG 、 ΔB が、それぞれ2, 2, 3ビット以下で表現できる場合には1バイト差分符号による圧縮が可能と判断し、差分 ΔR 、 ΔG 、 ΔB を設定した図3(c)の1バイト差分符号をデータ出力手段13に出力する(S18)。また、差分 ΔR 、 ΔG 、 ΔB が、それぞれ5, 4, 5ビット以下で表現できる場合は2バイト差分符号による圧縮が可能と判断し、差分 ΔR 、 ΔG 、 ΔB を設定した図3(d)の2バイト差分符号をデータ出力手段13に出力する(S18)。さらに、差分 ΔR 、 ΔG 、 ΔB が、それぞれ7, 6, 7ビット以下で表現できる場合は3バイト差分符号による圧縮が可能と判断し、差分 ΔR 、 ΔG 、 ΔB を設定した図3(e)の3バイト差分符号をデータ出力手段13に出力する(S18)。それ以外の場合、つまり差分 ΔR 、 ΔG 、 ΔB のそれぞれが7, 6, 7ビット以下で表現できない場合は差分符号による圧縮は不可能と判断し、今回の画素値そのものを設定した図3(f)の即値符号をデータ出力手段13に出力する(S19)。

【0042】そして、差分符号、即値符号を上述のように出力した場合には、今回の画素値を直前データ記憶手段17に格納すると共に(S6)、過去データ記憶手段18の次のインデックス番号のエントリに格納する(S7)。そして、ステップS2に戻って次の画素値を入力

する。

【0043】以上のような処理が繰り返され、画像ファイル2中の最後の画素まで処理を終えると(S3でYES)、連続数カウンタ14aの値が0より大きければ図3(a)のレンジス項にカウンタ値-1の値を設定した一致符号を出力し(S20, S21)、圧縮処理を終了する。

【0044】図5に、幾つかの画像について本実施例によるデータ圧縮方法および従来方法を使って実際に圧縮した場合の圧縮率の測定例を示す。従来方法としてはpackbits, LZW法, JPEG(ロスレスモード)を用いた。図中の数値は、「圧縮後のデータ量/元のデータ量」を示し、数値が小さいほど圧縮率が高いことを示す。また、1.00は圧縮不能を示す。ここで、画像1は文字列が表示された2つのウィンドウと電卓などのイメージ図が表示されているコンピュータ画面上の画像であり、画像2はコンピュータ画面上に表示された或るホームページ画面の画像であり、画像3は開かれた本の上に懐中時計が蓋を開いた状態で置かれた或るPhotoDisc中の画像であり、画像4は手前に湖、奥に森林のある風景を撮影した自然画であり、画像5は筋状の紋様が多数描かれたパターン画像である。この図5に示されるように本実施例のデータ圧縮方法によれば、様々な種類の画像に対して1番目か或いは2番目に高い圧縮率で圧縮することができる。

【0045】以上の実施例においては、圧縮符号の適用優先順序として、一致符号を最優先し、次にインデックス符号を優先し、差分符号は最後に適用した。これは、本実施例の場合、一致符号が1バイト、インデックス符号が1バイト、差分符号が最大3バイトであるため、この適用優先順序がより情報量を少なくし得る順序であるからである。しかし、各符号のバイト数は上述の例に限定されず、各種のものを採用することができ、その場合、圧縮符号の適用優先順序も変わり得る。例えば、レンジス項を13ビットにした2バイトの一致符号を用いる場合、同じ画素値が2個だけ連続する箇所では、若し、インデックス符号或いは1バイト差分符号による表現が可能であればその方が情報量が少ないので、一致符号よりインデックス符号或いは1バイト差分符号が優先的に適用される。

【0046】次に上述の実施例のデータ圧縮方法によって圧縮された画像を伸張し、元の画像を復元するデータ伸張方法の実施例について説明する。

【0047】図6は本発明を適用したデータ伸張装置の一例を示すブロック図である。この例のデータ伸張装置5は、制御手段51と、データ入力手段52と、データ出力手段53と、連続画素出力手段54と、過去データ出力手段55と、差分データ出力手段56と、セレクト59-1, 59-2といった機能手段を有すると共に、直前データ記憶手段57と過去データ記憶手段58といっ

た記憶手段を有している。また、圧縮ファイル3、復元ファイル6および利用者入出力装置7に接続されている。

【0048】データ入力手段52は、復元対象とする圧縮データを圧縮ファイル3から入力する手段である。前述したように圧縮データは、1バイトの一致符号(図3(a))、1バイトのインデックス符号(図3(b))、1バイト、2バイト及び3バイトの差分符号(図3(c)~(e))、4バイトの即値符号(図3(f))の合計6種類の圧縮符号から構成されている。データ入力手段52は圧縮データの先頭から順にこれらの圧縮符号を入力する。符号の区切りはバイト単位であり、何バイト符号であるかは符号の識別子ビットで判別可能である。つまり、読み込んだ1バイトの7ビット目が0であれば1バイト差分符号であり、7、6ビット目が1、0であれば2バイト差分符号であり、7、6、5ビット目が1、1、0であれば1バイトの一致符号であり、7、6、5、4ビット目が1、1、1、0であれば3バイト差分符号であり、7、6、5、4、3ビット目が1、1、1、1、0であればインデックス符号であり、7、6、5、4、3、2、1、0が1、1、1、1、1、0、0、0であれば即値符号である。なお、復元対象とする圧縮データの入力元はファイルに限られない。

【0049】直前データ記憶手段57は、直前に復元した画素値を保持する手段である。

【0050】過去データ記憶手段58は、過去に復元した画素値を複数種類記憶する手段であり、その構造およびそれへの画素値の格納方法はデータ圧縮装置側の過去データ記憶手段18と同一である。つまり、本実施例の場合、図2に示すように最大8個の画素値をデータ圧縮装置と同様の順序で記憶していく。

【0051】連続画素出力手段54は、一致符号の伸張手段であり、直前データ記憶手段57に記憶されている画素値を、データ入力手段52で入力された一致符号のレンジ項の値+1だけ連続して出力する。

【0052】過去データ出力手段55は、インデックス符号の伸張手段であり、データ入力手段52で入力されたインデックス符号が示すインデックス番号に対応して過去データ記憶手段58に記憶されている画素値を読み出して出力する。

【0053】差分データ出力手段56は、差分符号の伸張手段であり、データ入力手段52で入力された差分符号中の ΔR 、 ΔG 、 ΔB を直前データ記憶手段57に記憶されている画素値の各色値R、G、Bに加算して画素値を復元し、出力する。

【0054】セレクト59-1は、連続画素出力手段54、差分データ出力手段56および制御手段51から出力された画素値の何れかを、制御手段51からの制御信号に基づいて選択してセレクト59-2に出力すると共

に、過去データとして格納するために過去データ記憶手段58に出力する。

【0055】セレクト59-2は、セレクト59-1および過去データ出力手段55から出力された画素値の何れかを、制御手段51からの制御信号に基づいて選択してデータ出力手段53に出力すると共に、直前の画素値として格納するために直前データ記憶手段57に出力する。

【0056】データ出力手段53は、復元した画素値を復元ファイル6に出力する手段である。なお、復元データの出力先はファイルに限られない。

【0057】制御手段51は、データ伸張装置5全体を制御する手段である。

【0058】図7は図6のデータ伸張装置5の処理例を示すフローチャートである。以下、図6及び図7を参照して本実施例の動作を説明する。なお、データ伸張装置5をCPUおよびメモリを有するコンピュータで実現する場合、図7に示される処理を記述した伸張プログラムがCD-ROM、半導体メモリ、磁気ディスク等の機械読み取り可能な記録媒体(図6のM2)に記録されており、その記録媒体M2の伸張プログラムがコンピュータに読み取られ、コンピュータの動作を制御することにより、そのコンピュータを図6のデータ伸張装置5として機能させる。

【0059】復元対象となる圧縮データを格納する圧縮ファイル3、復元画像を格納する復元ファイル6を指定した伸張要求が利用者入出力装置7からデータ伸張装置5に入力されると、データ伸張装置5の制御手段51の制御の下で、図7に示す処理が開始される。

【0060】まず制御手段51は、装置各部を初期化する(S31)。この処理で、直前データ記憶手段57及び過去データ記憶手段58中の画素値がすべて0に初期化される。

【0061】次に制御手段51は、データ入力手段52を用いて圧縮ファイル3から最初の圧縮符号を入力し(S32)、その符号の種類を判定する(S34~S36)。

【0062】圧縮符号が即値符号の場合(S36でNO)、即値符号に設定された画素値そのものを制御手段51からセレクト59-1、59-2を通じてデータ出力手段53に送り、復元ファイル6に出力する(S37)。そして、その画素値を直前画素値として直前データ記憶手段57に記憶すると共に(S38)、過去データ記憶手段58に過去データの1つとして記憶し(S39)、ステップS32に戻って次の圧縮符号を入力する。

【0063】圧縮符号が一致符号の場合(S34でYES)、制御手段51はその一致符号を連続画素出力手段54に伝達し、連続画素出力手段54は直前データ記憶手段57に記憶されている画素値を、その一致符号のレ

ングス項の値+1の個数だけ連続してセレクト59-1に出力し、制御手段51はその出力をセレクト59-1、59-2で選択せしめてデータ出力手段53を通じて復元ファイル6に出力する(S40)。そして、その画素値を直前画素値として直前データ記憶手段57に記憶すると共に(S38)、過去データ記憶手段58に過去データの1つとして記憶し(S39)、ステップS32に戻って次の圧縮符号を入力する。

【0064】圧縮符号がインデックス符号の場合(S35でYES)、制御手段51はそのインデックス符号を過去データ出力手段55に伝達し、過去データ出力手段55はそのインデックス符号に設定されたインデックス番号に対応して過去データ記憶手段58に記憶されている画素値を読み取ってセレクト59-2に出力し、制御手段51はその出力をセレクト59-2で選択せしめてデータ出力手段53を通じて復元ファイル6に出力する(S41)。そして、その画素値を直前画素値として直前データ記憶手段57に記憶し(S42)、ステップS32に戻って次の圧縮符号を入力する。

【0065】圧縮符号が差分符号の場合(S36でYES)、制御手段51はその差分符号を差分データ出力手段56に伝達し、差分データ出力手段56は直前データ記憶手段57に記憶されている画素値のR、G、B値に、その差分符号の ΔR 、 ΔG 、 ΔB を加算した画素値をセレクト59-1に出力し、制御手段51はその出力をセレクト59-1、59-2で選択せしめてデータ出力手段53を通じて復元ファイル6に出力する(S43)。そして、その画素値を直前画素値として直前データ記憶手段57に記憶すると共に(S38)、過去データ記憶手段58に過去データの1つとして記憶し(S39)、ステップS32に戻って次の圧縮符号を入力する。

【0066】以上のような処理が繰り返され、圧縮ファイル3中の最後の圧縮符号まで処理を終えると(S33でYES)、伸張処理を終了する。

【0067】図8は本発明を適用したデータ圧縮装置の別の例を示すブロック図、図9はその処理例を示すフローチャートである。本実施例のデータ圧縮装置1が図1のデータ圧縮装置と相違するところは、圧縮対象画像の1ライン分の画素値を処理終わるごとに、直前の画素値や過去データなどを初期化することにより、圧縮ファイル3の伸張時に後述するように圧縮データの途中からの伸張処理を高速に行えるようにした点にある。このため制御手段11内には現在まで処理した画素数をカウントする画素数カウンタ11aが設けられており、また、ステップS1とステップS2との間にステップS22~S26が挿入されている等の変更が加えられている。なお、データ圧縮装置1をCPUおよびメモリを有するコンピュータで実現する場合、図9に示される処理を記述した圧縮プログラムがCD-ROM、半導体メモリ、磁

気ディスク等の機械読み取り可能な記録媒体(図8のM1)に記録されており、その記録媒体M1の圧縮プログラムがコンピュータに読み取られ、コンピュータの動作を制御することにより、そのコンピュータを図8のデータ圧縮装置1として機能させる。

【0068】以下、図1の実施例のデータ圧縮装置との差異を中心に本実施例の動作を説明する。

【0069】圧縮対象となる画像データを格納する画像ファイル2、圧縮画像を格納する圧縮ファイル3および画像の1ライン当たりの画素数Lを指定した圧縮要求が利用者入出力装置4からデータ圧縮装置1に入力されると、データ圧縮装置1の制御手段11の制御の下で、図9に示す処理が開始される。なお、画素数Lは画像ファイル3中の画像を走査して自動的に取得するようにしても良い。

【0070】まず制御手段11は、装置各部を初期化する(S1)。この処理では、直前データ記憶手段17及び過去データ記憶手段18中の画素値がすべて0に初期化され、直前データ比較手段14内の連続数カウンタ14aも0に初期化されると共に、画素数カウンタ11aが0に初期化される。

【0071】次に制御手段11は、データ入力手段12を用いて画像ファイル2から順次に画素値を入力して図1のデータ圧縮装置と同様の処理を繰り返すが(S2~S21)、本実施例では、1つの画素値を処理する前に、画素数カウンタ11aを1だけインクリメントすることで(S22)、初期化後から現在までに処理し終えた画素の数をカウントする。そして、画素数カウンタ11aの値がL+1に達していた場合(S23でYES)、1ライン分の処理を終えているので、連続数カウンタ14aの値が0より大きければ図3(a)のレンジス項にカウント値-1の値を設定した一致符号をデータ出力手段13に出力する処理を行った後(S24、S25)、また連続数カウンタ14aの値が0であれば処理S25をスキップして、それぞれステップS26に進み、直前データ記憶手段17及び過去データ記憶手段18中の画素値をすべて0に初期化し、直前データ比較手段14内の連続数カウンタ14aを0に初期化し、また画素数カウンタ11aを1に設定する。そして、ステップS2に進んで次の画素値を処理する。

【0072】このように画像の1ライン分の画素値を処理し終える毎に再度初期化することにより、圧縮率は多少劣化することになるが、圧縮データの途中からの伸張処理が高速に行える。以下、伸張時の処理について説明する。

【0073】図10は図8のデータ圧縮装置で生成された画像ファイル3中の圧縮画像を伸張するデータ伸張装置の一例を示すブロック図である。本実施例のデータ伸張装置5が図6のデータ伸張装置と相違するところは、利用者入出力装置7からの伸張要求時、復元を開始した

いラインが指定された場合に、そのラインの頭出しを行い、そのライン以降の画素値から実際に復元を行う機能を有する点にある。このため制御手段 1 1 内には、画素カウンタ 5 1 a、ラインカウンタ 5 1 b および頭出し手段 5 1 が設けられており、またデータ伸張装置の処理も図 1 1 および図 1 2 に示すように変更されている。なお、データ伸張装置 5 を CPU およびメモリを有するコンピュータで実現する場合、図 1 1 および図 1 2 に示される処理を記述した伸張プログラムが CD-ROM、半導体メモリ、磁気ディスク等の機械読み取り可能な記録媒体 (図 1 0 の M2) に記録されており、その記録媒体 M2 の伸張プログラムがコンピュータに読み取られ、コンピュータの動作を制御することにより、そのコンピュータを図 1 0 のデータ伸張装置 5 として機能させる。

【0074】以下、図 6 の実施例のデータ伸張装置との差異を中心に本実施例の動作を説明する。

【0075】復元対象となる圧縮データを格納する圧縮ファイル 3、復元画像を格納する復元ファイル 6 および 1 ライン当たりの画素数 L 並びに必要に応じて復元開始ライン番号 m を指定した伸張要求が利用者入出力装置 7 からデータ伸張装置 5 に入力されると、データ伸張装置 5 の制御手段 5 1 の制御の下で、図 1 1 に示す処理が開始される。なお、1 ライン当たりの画素数は圧縮時に圧縮ファイル 3 中に記録しておき、これを読み込むようにしても良い。また、復元開始ライン番号 m は先頭ラインを 1 番として指定する。例えば 3 番目のラインの場合、m = 3 とする。

【0076】まず制御手段 5 1 は、装置各部を初期化する (S 5 1)。この処理で、直前データ記憶手段 5 7 及び過去データ記憶手段 5 8 中の画素値がすべて 0 に初期化されると共に、画素数カウンタ 5 1 a およびラインカウンタ 5 1 b も 0 に初期化される。

【0077】次に制御手段 5 1 は、復元開始ライン番号の指定があったか否かを判定し (S 5 2)、そのような指定が有れば、頭出し手段 5 1 c によってステップ S 5 3 ~ S 6 1 の処理を実行した後図 1 2 の処理へ進み、そのような指定が無ければ直ちに図 1 2 の処理へ進む。

【0078】頭出し手段 5 1 c による頭出し処理は、データ入力手段 5 2 により圧縮ファイル 3 の先頭の圧縮符号から順に 1 符号ずつ読み出し (S 5 3)、その圧縮符号が一致符号であれば (S 5 5 で YES)、その一致符号のレングス項の値 + 1 の回数だけ画素数カウンタ 5 1 a をインクリメントし (S 5 6)、それ以外の圧縮符号であれば画素数カウンタ 5 1 a を 1 だけインクリメントしていく (S 5 7)。そして、インクリメント後の画素数カウンタ 5 1 a のカウント値が 1 ライン分の画素数 L - 1 に達したら、ラインカウンタ 5 1 b を 1 だけインクリメントしていく (S 5 8、S 5 9)、ラインカウンタ 5 1 b のカウント値が復元開始ライン番号 m - 1 に達したら (S 6 0 で YES)、頭出し完了となる。符号の種

類を判別し復元される画素数を計算するだけで、実際の画素値の復元は行わないので、頭出しは高速に行える。なお、ステップ S 6 0 で YES と判断される前に、最後の圧縮符号まで処理されたときは (S 5 4 で YES)、復元開始ライン番号の指定に誤りがあったことになり、利用者入出力装置 7 にエラーメッセージを出力する (S 6 1)。

【0079】図 1 2 の処理は、図 7 の処理とほぼ同じである。但し、圧縮時、1 ライン毎に直前の画素値および過去データが初期化されているため、伸張時も、画素カウンタ 5 1 a を用いて復元画素数をカウントし (S 7 1、S 7 3)、1 ラインの最後の画素まで復元し終えたら (S 7 2、S 7 4 で YES)、ステップ S 3 1 に戻って、直前データ記憶手段 5 7 及び過去データ記憶手段 5 8 中の画素値をすべて 0 に初期化すると共に、画素数カウンタ 5 1 a を 0 に初期化する。

【0080】図 8 および図 1 0 の実施例では、任意のラインの先頭からの復元を可能としたが、任意のラインの任意の行からの復元を可能にすることもできる。その場合、復元開始ライン番号以外に復元開始行番号を指定し、指定された復元開始ラインに頭出した後、復元開始行の直前までの画素値は出力せずに直前画素値および過去データとして記憶するのに止め、復元開始行の画素から実際に復元ファイル 6 に出力する通常処理を行う。

【0081】

【発明の効果】以上説明したように本発明によれば以下のような効果を得ることができる。

【0082】一致符号、差分符号、インデックス符号および即値符号の合計 4 種類の符号を使い分け、圧縮対象となるデータ群の先頭から順に 1 データ毎に、圧縮後のデータ量がより少なくなる符号を選択してデータの圧縮を行うため、様々な種類の画像に対して従来方法のうち最も高い圧縮率となる圧縮方法と遜色ない圧縮率を得ることができる。

【0083】全ての種類の符号をバイト単位の符号とする構成にあつては、ビット単位でのデータ処理が不要になり、その分、処理量が低減する。

【0084】予め定められた画素数の処理を終える毎に直前の画素値や過去データなどを初期化する構成にあつては、伸張時、圧縮データの途中からの伸張処理を高速に行うことができる。

【図面の簡単な説明】

【図 1】本発明を適用したデータ圧縮装置の一例を示すブロック図である。

【図 2】過去データ記憶手段の構成例を示す図である。

【図 3】本発明の実施例で使用される圧縮符号のフォーマット例である。

【図 4】図 1 のデータ圧縮装置の処理例を示すフローチャートである。

【図 5】幾つかの画像について本発明の実施例によるデ

ータ圧縮方法および従来方法を使って実際に圧縮した場合の圧縮率の測定例を示す図である。

【図 6】本発明を適用したデータ伸張装置の一例を示すブロック図である。

【図 7】図 6 のデータ伸張装置の処理例を示すフローチャートである。

【図 8】本発明を適用したデータ圧縮装置の別の例を示すブロック図である。

【図 9】図 8 のデータ処理装置の処理例を示すフローチャートである。

【図 10】本発明を適用したデータ伸張装置の別の例を示すブロック図である。

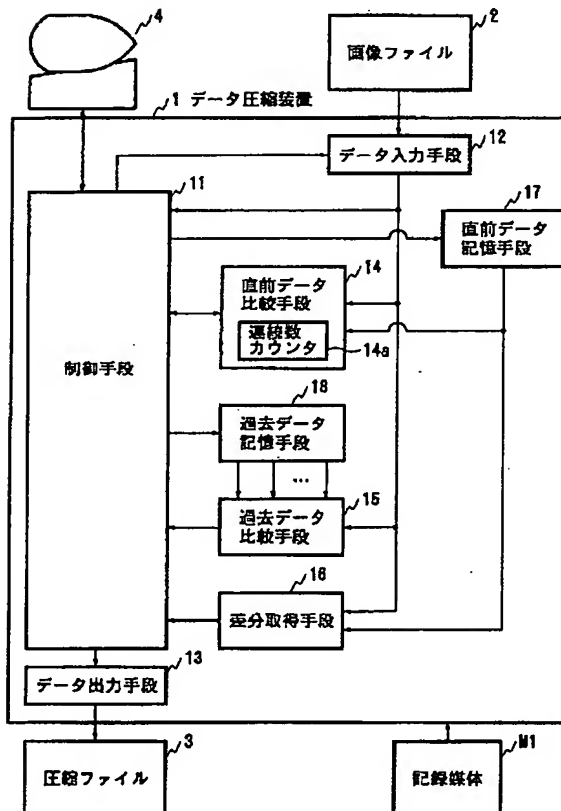
【図 11】図 10 のデータ伸張装置の処理例を示すフローチャートである。

【図 12】図 10 のデータ伸張装置の処理例を示すフローチャートである。

【符号の説明】

- 1…データ圧縮装置
- 11…制御手段
- 12…データ入力手段
- 13…データ出力手段
- 14…直前データ比較手段
- 14a…連続数カウンタ
- 15…過去データ比較手段
- 16…差分取得手段
- 17…直前データ記憶手段
- 18…過去データ記憶手段
- 2…画像ファイル
- 3…圧縮ファイル
- 4, 7…利用者入出力装置
- 5…データ伸張装置
- 6…復元ファイル

【図 1】



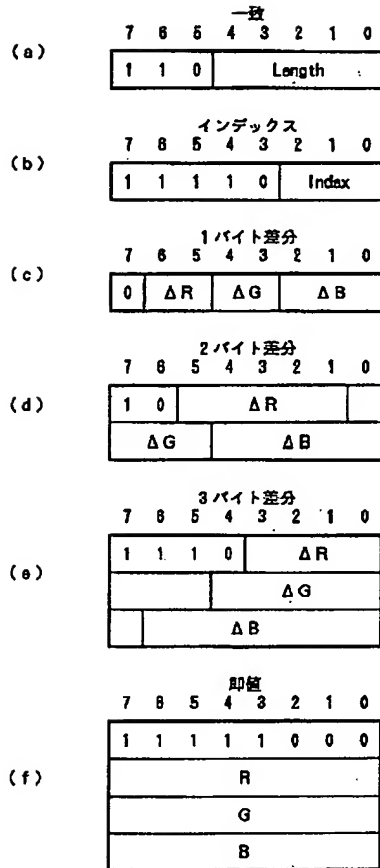
【図 2】

| Index | 画素値 | |
|-------|-----|------|
| 000 | | ～ E0 |
| 001 | | ～ E1 |
| 010 | | ～ E2 |
| 011 | | ～ E3 |
| 100 | | ～ E4 |
| 101 | | ～ E5 |
| 110 | | ～ E6 |
| 111 | | ～ E7 |

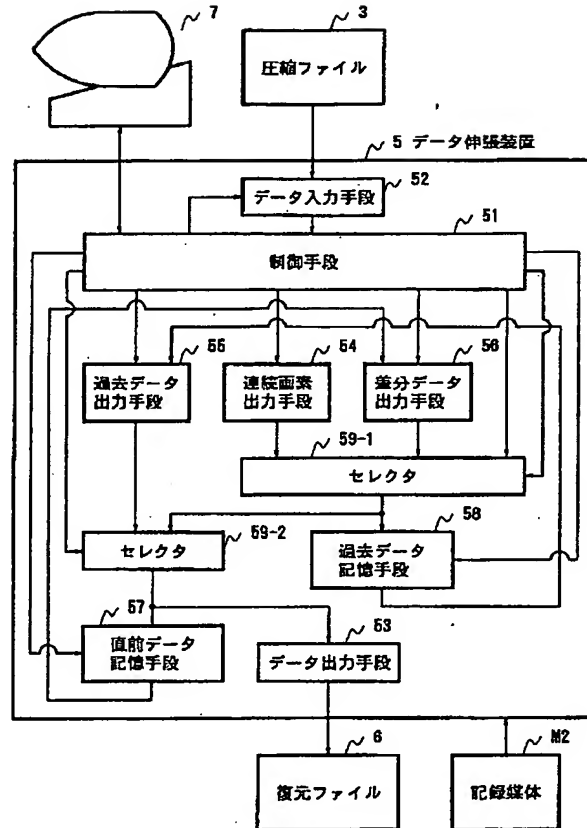
【図 5】

| 圧縮方法 画像 | 本発明 | Packbits | LZ法 | JPEG (ロスレスモード) |
|------------|------|----------|------|----------------|
| 1 | 0.09 | 0.21 | 0.03 | 0.39 |
| 2 | 0.19 | 0.37 | 0.12 | 0.31 |
| 3 | 0.56 | 0.89 | 0.92 | 0.59 |
| 4 | 0.87 | 1.00 | 1.00 | 0.81 |
| 5 | 0.40 | 1.00 | 0.52 | 0.41 |

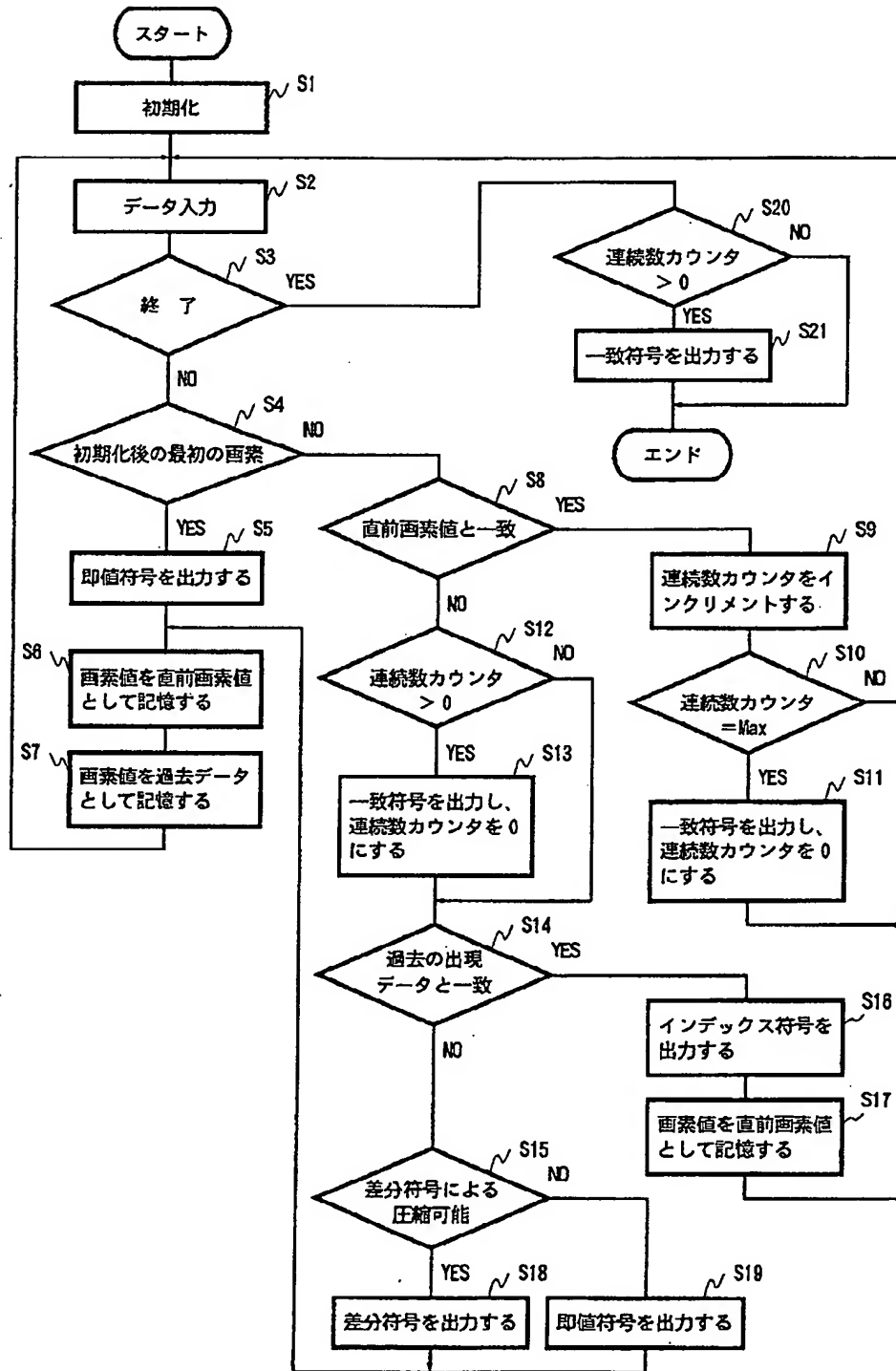
【図3】



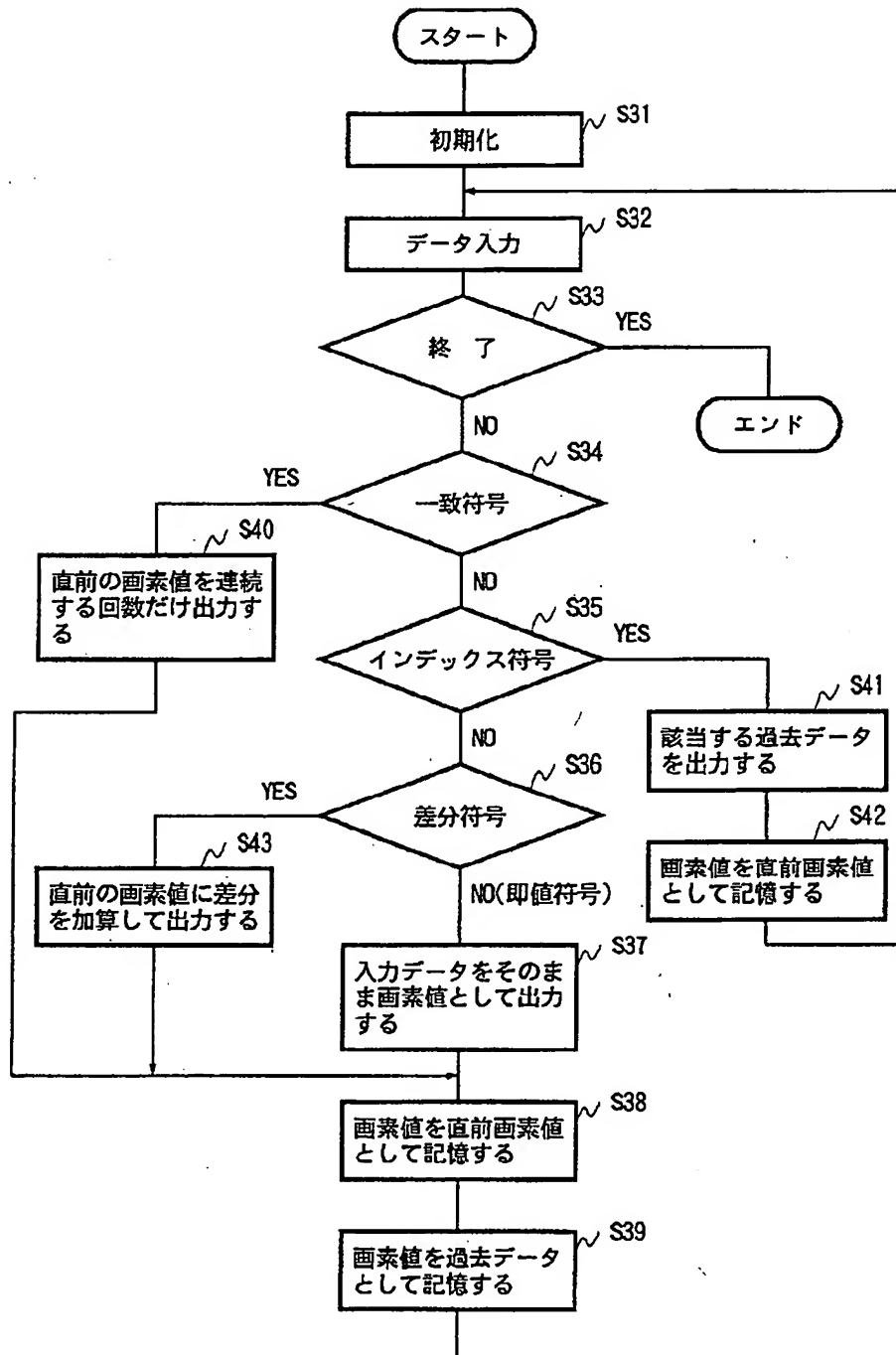
【図6】



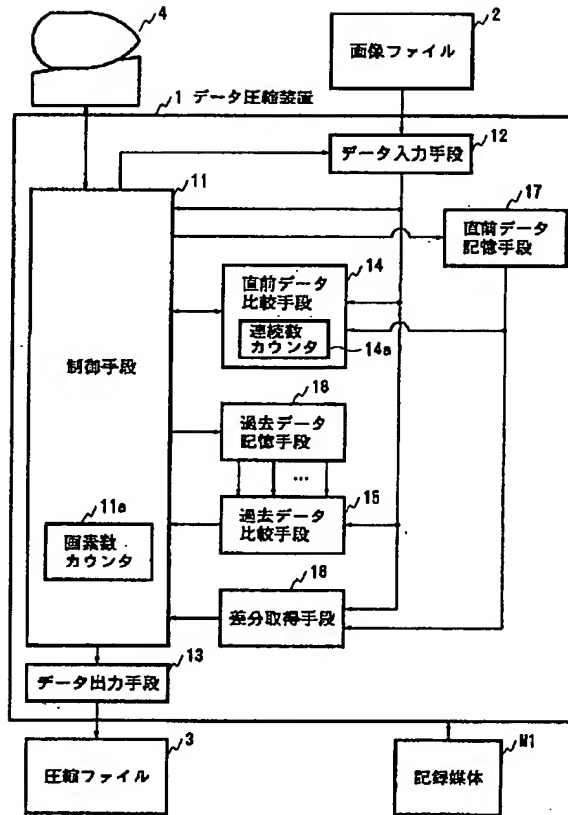
【図 4】



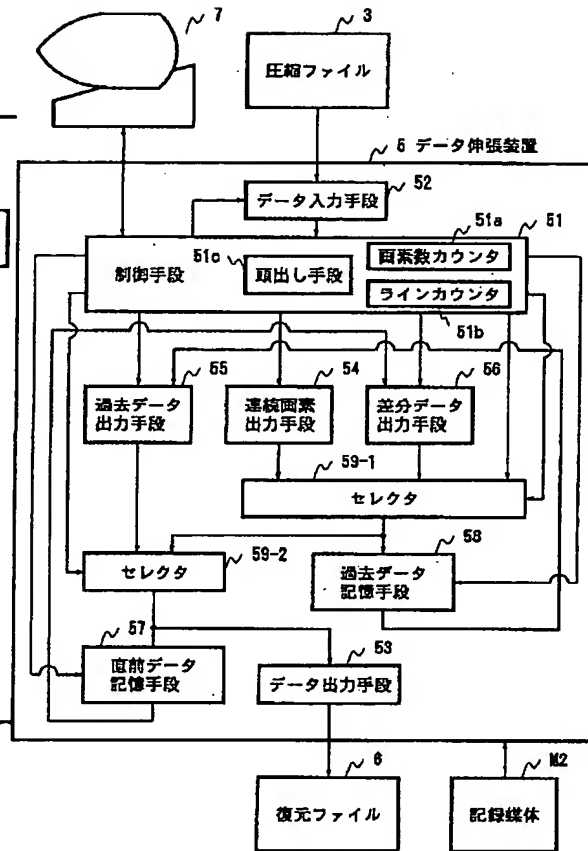
【図 7】



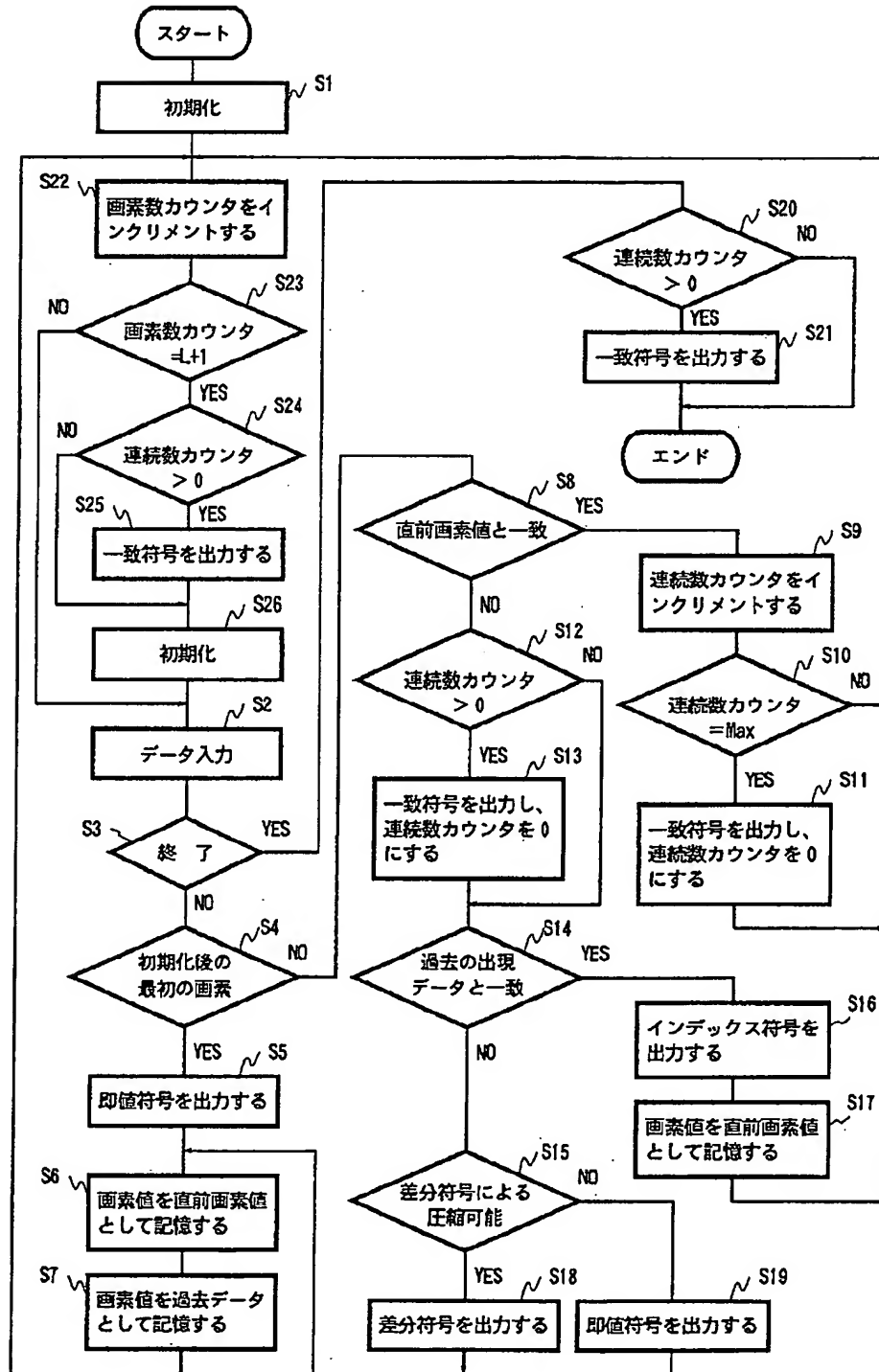
【図 8】



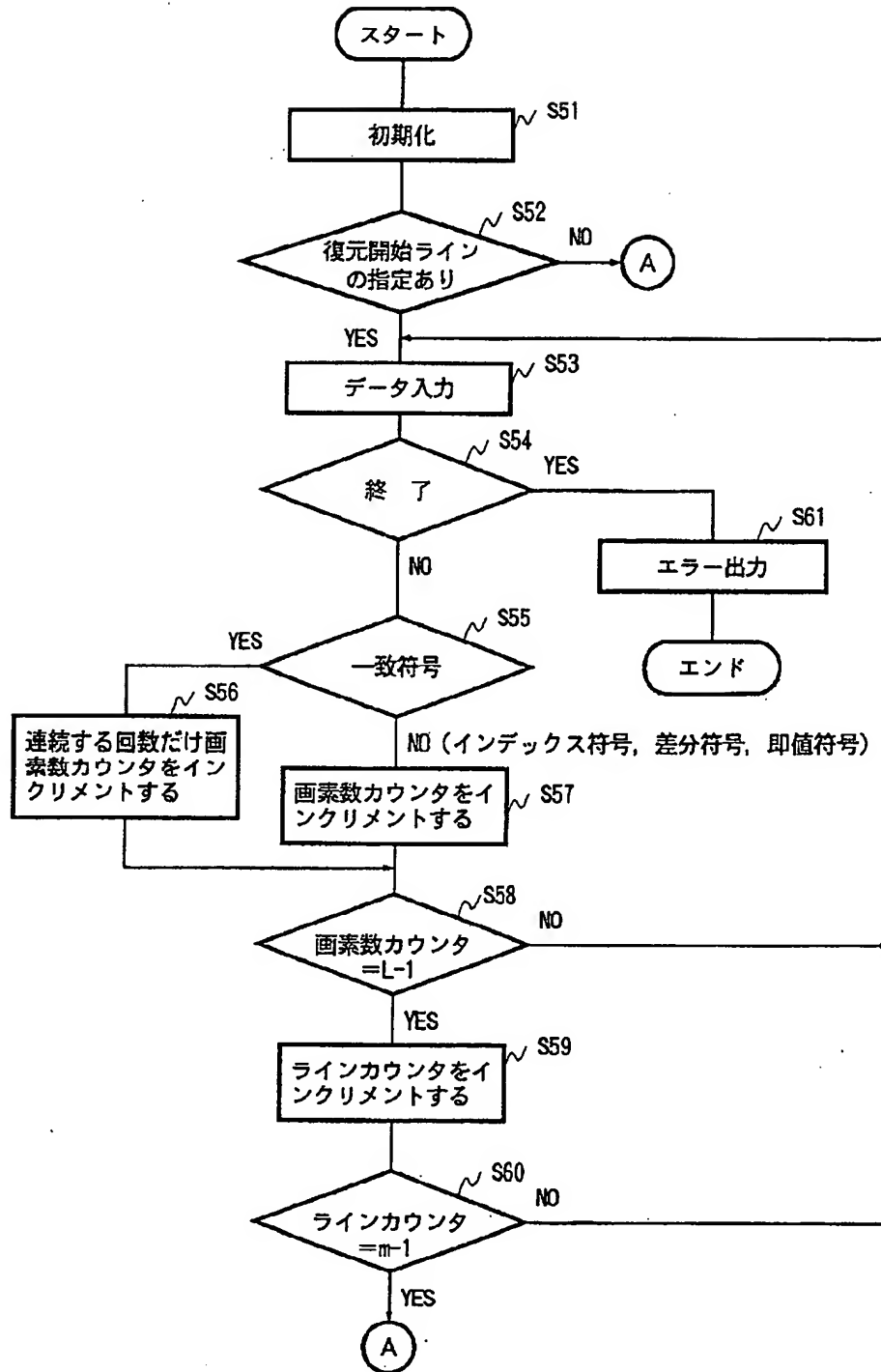
【図 10】



【図 9】



【図 1 1】



【図 12】

